



IES SERRA
PERENXISA



MORACORP ELECTRONICS

Implementación de un ERP para la Administración y Soporte de Productos Electrónicos



Autor: Álvaro Mora López - Tutor: Sergi García
GRADO SUPERIOR DE DESARROLLO DE APLICACIONES MULTIPLATAFORMA
MAYO – JUNIO 2022

ÍNDICE

Resumen	4
Castellano.....	4
Valenciano.....	4
Introducción	5
Sinopsis del Proyecto	5
Tecnologías del Proyecto	5
Justificación del uso de las Tecnologías	6
Simulación del Proyecto.....	8
Estado del Arte.....	9
Estudio de Viabilidad.....	9
Estudio del Mercado	9
DAFO y CAME.....	10
Viabilidad Técnica y Económica	11
Recursos de Hardware	11
Recursos de Software	11
Recursos Humanos.....	11
Viabilidad Temporal	11
Diseño	12
Diagrama Entidad Relación.....	12
ODOO	14
Módulos Básicos	14
Módulo de Sitio Web	14
Módulo de Contactos.....	15
Servidor de Correos ODOO	15
Pasos Previos.....	15
Implementación	16
Módulo de Administración Moracorp	19
Modelo de Productos.....	20
Relaciones del Modelo.....	20
Restricciones y Utilidades del Modelo.....	20
Funcionamiento del Modelo.....	21
Vistas del Modelo.....	22
Informe del Modelo	23
Modelo de Compras.....	24
Relaciones del Modelo.....	24
Restricciones y Utilidades del Modelo.....	24
Funcionamiento del Modelo.....	25
Vistas del Modelo.....	25

Informe del Modelo	26
Modelo de Ventas	27
Relaciones del Modelo de Ventas.....	27
Restricciones y Utilidades del Modelo.....	28
Funcionamiento del Modelo.....	28
Vistas del Modelo.....	29
Informe del Modelo	30
Modelo Stock	31
Relaciones del Modelo.....	31
Restricciones y Utilidades del Modelo.....	31
Funcionamiento del Modelo.....	32
Vistas del Modelo.....	32
Informe del Modelo	33
Modelo de Proveedores.....	34
Relaciones del Modelo.....	34
Restricciones del Modelo.....	34
Funcionamiento del Modelo.....	35
Vistas del Modelo.....	35
Modelo de Clientes	36
Relaciones del Modelo.....	36
Restricciones del Modelo.....	36
Funcionamiento del Modelo.....	37
Vistas del Modelo.....	37
Modelo de Pedidos	38
Relaciones del Modelo.....	38
Restricciones y Utilidades del Modelo.....	38
Funcionamiento del Modelo.....	39
Vistas del Modelo.....	39
Informe del Modelo	41
Modelo de Transacción de Venta	42
Relaciones del Modelo.....	42
Restricciones del Modelo.....	42
Funcionamiento del Modelo.....	42
Vistas del Modelo.....	43
Informes del Modulo	43
APIS	43
Quasar	46
Aplicación Moracorp.....	46
Librerías Implementadas en la Aplicación	46
Menú de la Aplicación.....	47

Sección de Productos	48
Sección de Compras	50
Sección de Ventas y Transacciones de Venta	52
Transacciones de Venta	53
Venta	54
Sección de Stock	56
Sección de Proveedores	59
Sección de Clientes	61
Sección de Pedidos	62
Aplicación de Escritorio	65
Despliegue	66
Diagrama de Despliegue	66
Conclusión	66
GitHub	67
Credenciales del Servidor OODOO	68
DockerHub	68
Glosario	69
Bibliografía	72

Resumen

Castellano

El proyecto **Moracorp Electronics** consistirá en la implementación y configuración de un **ERP**, el cual se encargará de centralizar y organizar los recursos empresariales. Este **ERP** permitirá realizar a la empresa una sencilla y eficiente gestión o administración de las labores de la empresa, como son el inventariado de productos, las compras y ventas de los artículos que distribuyen, el soporte sobre esos mismos artículos, la tramitación de los pedidos, la gestión de las comunicaciones y relaciones con los proveedores y clientes. Y además este **ERP** les permitirá llevar a cabo una eficiente tramitación y gestión de los documentos empresariales.

Toda esa eficiencia y sencilla administración se debe a que implementaré un **ERP** hecho a **la medida de la empresa Moracorp Electronics**, en el **ERP** se configurarán una serie de módulos básico que son proporcionados por la propia empresa distribuidora del **software**, estos módulos aportarán diversas funcionalidades y características que serán beneficiosas para cubrir las necesidades más básicas de la Empresa. Pero la parte fundamental de este **ERP** será la creación de un **módulo personalizado** que se encargará de administrar y organizar las labores primordiales de la empresa **Moracorp Electronics**, la cual se dedica a la compra y venta de productos electrónicos y de dar soporte sobre esos mismos productos.

Por último, en este proyecto se llevará a cabo el desarrollo de una **aplicación** que permitirá a los trabajadores de la empresa acceder de una forma sencilla e intuitiva a la información que está ubicada en el **ERP**. Esta aplicación permitirá interactuar y manipular de una manera eficiente y sencilla la información que hay en el **ERP** desde cualquier dispositivo electrónico.

Valenciano

El projecte **Moracorp Electronics** consistirà en la implementació i configuració d'un **ERP**, el qual s'encarregarà de centralitzar i organitzar els recursos empresarials. Este **ERP** permetrà realitzar a l'empresa una senzilla i eficient gestió o administració de les labors de l'empresa, com són l'inventariat de productes, les compres i vendes dels articles que distribuïxen, el suport sobre eixos mateixos articles, la tramitació de les comandes, la gestió de les comunicacions i relacions amb els proveïdors i clients. I a més este **ERP** els permetrà dur a terme una eficient tramitació i gestió dels documents empresarials.

Tota eixa eficiència i senzilla administració se deu al fet que implementaré un **ERP** fet a **la mesura de l'empresa Moracorp Electronics**, en l'**ERP** es configuraran una sèrie de mòduls bàsic que són proporcionats per la pròpia empresa distribuïdora del **software**, estos mòduls aportaren diverses funcionalitats i característiques que seran beneficioses per a cobrir les necessitats més bàsiques de l'Empresa. Però la part fonamental d'este ERP serà la creació d'un **mòdul personalitzat** que s'encarregarà d'administrar i organitzar les labors primordials de l'empresa **Moracorp Electronics**, la qual es dedica a la compra i venda de productes electrònics i de donar suport sobre eixos mateixos productes.

Finalment, en este projecte es durà a terme el desenrotllament d'una **aplicació** que permetrà als treballadors de l'empresa accedir d'una forma senzilla i intuïtiva a la informació que està ubicada en l'**ERP**. Esta aplicació permetrà interactuar i manipular d'una manera eficient i senzilla la informació que hi ha en l'**ERP** des de qualsevol dispositiu electrònic.

Introducción

Sinopsis del Proyecto

Este proyecto se basa en la implementación y configuración de un **ERP** (Enterprise Resource Planning, que significa en castellano **Sistema de Planificación de Recursos Empresariales**) con **ODOO** que permita la gestión de una empresa de electrónica que ofrece compra y venta de sus productos y da soporte sobre esos mismos productos. La idea principal de este proyecto es brindar a nuestro cliente un **ERP** hecho a la medida que centralice todas las herramientas que necesite dicha empresa para rentabilizar y optimizar su negocio.

En este **ERP** instalaremos módulos básicos de **ODOO** (Por Ejemplo: **Sitio Web, Compras, Empleados, Ventas, Puntos de Ventas, Facturación, Inventario, etc.**) para cubrir las necesidades más básicas de la Empresa. Pero la parte más importante de este será la creación de **un módulo hecho a la medida del cliente** que gestionará el stock de dispositivos, la compra y venta de los mismos. Este mismo modulo nos permitirá crear informes referentes a las compras, ventas y stocks. Y además gestionará a las empresas que dará soporte dicha empresa de tecnología con sus productos.

Aparte de todo lo dicho anteriormente también **desarrollare una aplicación con interfaz gráfica con el Framework front-end Quasar** que nos permita acceder a la información que hay en el servidor **ODOO** referente a las actividades de la Empresa de Electrónica. Esta interfaz gráfica será clara, concisa y muy intuitiva para que los empleados puedan navegar fácilmente por ella.

Tecnologías del Proyecto

Para la elaboración de este proyecto haremos uso de las siguientes **tecnologías o plataformas**:

- **Docker:** es un sistema de virtualización y gestor o administrador de contenedores de código abierto que utiliza las características del núcleo de Linux para permitir el desarrollo y despliegue de aplicaciones. Docker empaqueta software en unidades estandarizadas llamadas contenedores que incluyen todo lo necesario para que el software se ejecute, incluidas bibliotecas, herramientas de sistema, código y tiempo de ejecución. Con Docker, puede implementar y ajustar la escala de aplicaciones rápidamente en cualquier entorno con la certeza de saber que su código se ejecutará. Si estamos utilizando **Docker** podemos usar la plataforma **DockerHub** para subir a la nube la imagen de nuestro repositorio.
- **ODOO :** es un **software de gestión (ERP) de código abierto, compuesto por diferentes aplicaciones o módulos**, que permiten una gestión completa de las diferentes áreas de las empresas. Este software permite la instalación de los módulos que necesite la empresa, estos módulos pueden ser gratuitos o de pago. Estos módulos por norma general son independientes por lo que, si surgen nuevas necesidades, la empresa podrá ir incluyendo módulos de forma progresiva, según los vaya a necesitar su negocio.
- **PostgreSQL:** es un **poderoso sistema gestor de bases de datos relacionales**, este gestor está orientado a objetos, es multiplataforma y es de código abierto. Este gestor de base de datos cuenta con una larga carrera de desarrollo activo que le ha valido una sólida reputación por su confiabilidad y robustez de funciones y rendimiento.
- **Python:** es un **lenguaje de programación de alto nivel interpretado multiparadigma** que es utilizado para el desarrollo de aplicaciones o programas. Este lenguaje es multiplataforma y de código abierto que cuenta con una gran comunidad y apoyo por parte de los programadores. **Python** es un lenguaje de programación que posee una sintaxis sencilla que permite su fácil lectura, esa fácil lectura se ve beneficiada en gran medida por su tipado dinámico.

- **Quasar:** es un **front-end Framework** basado en [Vue.js](#) de código abierto con licencia del [MIT](#). Este framework nos brindara la oportunidad de crear de una forma rápida y eficiente sitios web y aplicaciones, con una sola base de código, e implementarlo en la web como **SPA** (Aplicación de una sola Página), **PWA** (Aplicación Web Progresiva), **SSR** (Aplicación renderizada del lado del Servidor), **BEX** (extensión del navegador), en una aplicación móvil, usando **Cordova** para **iOS** y **Android**, y en una aplicación de escritorio, utilizando **Electron** para **Mac, Windows y Linux**. Este **framework front-end** hace uso del **lenguaje de marcado HTML, del lenguaje de diseño o estilos CSS** y del lenguaje de programación **Javascript**.
- **JavaScript:** es un **lenguaje de programación de alto nivel interpretado** usado en el desarrollo de páginas y aplicaciones web. Complementa al tradicional **HTML** que escribe el contenido de las páginas en el navegador y les agrega dinamismo, interactividad, animación y efectos. **JavaScript** funciona desde el navegador, por lo que muchas funciones están disponibles en las páginas, aun sin conexión. También podemos utilizar **JavaScript** para el desarrollo de todo el **back-end** de una aplicación, programando con **Node.js**.
- **HTML:** es un **lenguaje de marcado** que sirve para desarrollar y definir el contenido de las páginas web. Se compone en base a etiquetas, también llamadas marcas o tags, con las cuales conseguimos expresar las partes de un documento, cabecera, cuerpo, encabezados, párrafos, etc. En definitiva, el contenido de una página web.
- **CSS:** es un **lenguaje de estilos o diseño** utilizado para describir la presentación y apariencia de documentos HTML o XML. Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML o XHTML. **CSS** es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web e **Interfaces Graficas de usuario (GUI)**.
- **Git:** es un sistema de gestión y control de versiones. Este sistema nos permitirá llevar a cabo la gestión de las distintas versiones de un proyecto. **Git** se verá apoyado por la plataforma en la nube [GitHub](#), [GitHub](#) nos permitirá subir nuestros proyectos a la nube donde podremos continuar con el control de versiones, y además el poder utilizar la plataforma [GitHub](#) nos brindará la posibilidad compartir código entre desarrolladores e incluso poder trabajar conjuntamente en un solo proyecto varios desarrolladores.

Justificación del uso de las Tecnologías

Utilizare **Docker** para virtualizar tanto el servidor **ODOO** como el gestor de base de datos **PostgreSQL**, cada uno de ellos en un contenedor. Hago esto porque cuando creo un nuevo proyecto en **ODOO**, de forma regular necesito realizar instalaciones de librerías y realizar configuraciones especiales en el entorno de desarrollo, además de tener proyectos de **ODOO** en diferentes versiones, las cuales deben coexistir. Con **Docker** puedo crear un archivo de configuración que permita preparar un contenedor que contenga todas las dependencias y configuraciones que el proyecto necesite. A la vez, puedo compartir esta configuración con otros desarrolladores, y estos podrán importar el proyecto sin configuraciones ni modificaciones adicionales. Básicamente gracias a **Docker** podre tener en un **mismo ordenador cualquier versión de OODOO y tener múltiples instancias corriendo, lo que facilitará el desarrollo y la puesta en marcha en cualquier momento, y sobre todo en cualquier entorno**.

He escogido **ODOO** para la implementación y configuración del **ERP**. Como sistema **ERP** he decidido utilizar **ODOO** porque es de código abierto y gratuito. La versión que vamos a usar para el desarrollo del proyecto es la versión gratuita, la cual se denomina comunitaria que está bajo la licencia **LGPLv3**, esta versión nos permitirá acceder a gran parte de los módulos que ofrece la empresa belga creadora de **ODOO Odoo S.A**, aunque el soporte y ciertos módulos muy importantes para las empresas son de pago. En lo referido a la versión del software de **ODOO**, utilizaremos la **versión 14**, ya sé que ha salido una versión del software más moderna como es la **versión 15** pero no me he planteado usarla para la elaboración del proyecto por los posibles problemas que puede dar un software nuevo que apenas ha sido testeado por la comunidad de usuarios de **ODOO**.

Al ser **ODOO** de código abierto cuenta con una gran comunidad y de muchos foros, los cuales nos proporcionan información sobre **ODOO** y nos pueden resolver las posibles dudas que tengamos, gracias a todo ello tenemos mucha documentación e información que nos puede servir de ayuda. Y también he seleccionado dicho **ERP** porque tengo

experiencia con él porque es el **ERP** que hemos estado usando en la asignatura de Sistemas de Gestión Empresarial a lo largo del curso.

Otra ventaja que nos brinda **ODOO** es que no necesita una gran potencia en el servidor o dispositivo donde se vaya a instalar, esto quiere decir que los requerimientos de hardware mínimos que necesita el servidor **ODOO** para funcionar correctamente son bastante asequibles, porque solo necesita para funcionar un procesador **dual Core ,2 GB RAM** pero lo más recomendable sería contar con **8 GB de RAM**, un **disco duro** con una capacidad mínima de **50 GB** y también necesitaría una **tarjeta de red** (para permitir las conexiones). Con estos requerimientos mínimos solo serían útiles para una empresa muy pequeña porque solo podrían trabajar con ese **ERP** unas 5 personas, habría que mejorar el hardware para hacer más escalabre el servidor **ODOO** y poder trabajar múltiples personas sobre él.

Para implementar un **ERP** necesitamos un gestor de base de datos para ello utilizaremos como gestor de base de datos **PostgreSQL** para gestionar y administrar la base de datos del **ERP** implementado con **ODOO**. Utilizare este gestor de base de datos por su gran robustez, tolerancia fallos y consistencia, y además porque es capaz de funcionar de manera estable en el servidor.

Y utilizaré el lenguaje interpretado **Python** para desarrollar mi modulo que estará hecho a la medida de la empresa. Hago uso de Python porque es el lenguaje de programación que se utiliza para desarrollar módulos en **ODOO**. **ODOO** solo soporta el lenguaje **Python y JavaScript**.

He utilizado **Quasar como front-end framework** para desarrollar la aplicación grafica de escritorio porque es un **framework** eficiente y está muy bien optimizado para desarrollar aplicaciones de alto rendimiento en un breve periodo de tiempo. También he seleccionado este **framework** porque es de código libre y gratuito. Otras de las razones por la que he elegido **Quasar** es que cuenta con una gran comunidad y de muchos foros, los cuales nos proporcionan información y ayuda. Y también he elegido este **framework** por su sencilla curva de aprendizaje porque con unos conocimientos básicos en **JavaScript, HTML y CSS** te puedes montar una aplicación bastante cumplidora en términos de diseño y funcionalidad. Y el ultimo motivo que he tenido para elegir **Quasar** es que cuenta con una amplia variedad de componentes y herramientas, nosotros como desarrolladores no necesitamos bibliotecas de terceros como **Bootstrap o Hammer.js** con los mismo componentes y herramientas que nos brinda **Quasar** nos son más que suficientes para poder desarrollar nuestros proyectos.

Para implementar **Quasar** hare uso del **lenguaje de programación JavaScript** el cual me permitirá dar funcionalidad a la aplicación, y también hare uso del **lenguaje de marcado HTML** y del **lenguaje de estilos CSS** para diseñar una aplicación de escritorio atractiva, intuitiva y bien diseñada.

En este proyecto hare uso del sistema de control de versiones **Git** y de la aplicación en la nube **Github**, estas tecnologías me permitirán llevar a cabo un control de versiones del módulo personalizado que voy a patentar para el **ERP ODOO** y también utilizare estas tecnologías para llevar a cabo un control de versiones de la aplicación de escritorio que voy a hacer con el **Framework Quasar**.

Simulación del Proyecto

Todo este proyecto va a ser desarrollado en una máquina virtual con el sistema operativo Linux cuya versión es **Mint Mate 20.3 Una**, para llevar a cabo la virtualización de ese sistema operativo vamos a hacer uso del software de virtualización [VirtualBox](#). El hardware que va a utilizar la máquina virtual es la siguiente:

REQUISITOS				
CARACTERISTICAS	PROCESADOR	RAM	DISCO DURO	ADAPTADOR DE RED
UTILIZADAS EN EL PROYECTO	Procesador I7-8750H 2,20GHz y he asignado 2 núcleos a la máquina virtual	5GB	150 GB	Adaptador Intel PRO/1000 MT

Por último, en este punto veremos las **características de las tecnologías y plataformas** que hemos utilizado para elaborar el proyecto.

PLATAFORMAS Y TECNOLOGIAS		
CARACTERISTICAS	Nombre	Versión
UTILIZADAS EN EL PROYECTO	Docker	20.10.4
	VirtualBox	6.1.32
	ODOO	14
	PostgreSQL	10
	Python	3.8.10
	Quasar	2.6.6
	JavaScript	1.7
	HTML	5
	CSS	3

Estado del Arte

Tras el estudio exhaustivo y análisis de otro software he llegado a la conclusión que este proyecto se tiene que cimentar en **cuatro pilares fundamentales la eficiencia, portabilidad, automatización y la interoperabilidad**. A esta conclusión he llegado gracias en gran parte por las **FCTS**, porque en dichas **FCTS** he estado trabajando con el **ERP de Windows denominado Business Central**, al estar trabajando con dicho **ERP** en la empresa me ha permitido conocer cómo funciona de forma básica el **ERP** y como programar en él, y además me ha permitido obtener conocimientos de lo que en realidad desea un cliente de un **Sistema de Planificación de Recursos Empresariales (ERP)**. Además, gracias a las **FCTS** he podido comparar el **ERP de OODOO** que es que voy a usar en la elaboración del proyecto con su competencia directa como es el **ERP de Windows Business Central**.

Un cliente quiere **eficiencia**, busca que el **ERP** le permita hacer cualquier cosa de **gestión y administración empresarial** ya sea **registrar información de todo tipo** en el **ERP** y **crear documentación como albaranes, facturas, informes y códigos de barras**. También el cliente quiere **diversas políticas que restrinjan el acceso a la información dependiendo de la labor** que hagas y a qué **departamento** pertenezcas. El cliente también busca mantener un **control sobre los usuarios que acceden al ERP y saber las modificaciones o posibles cambios que hayan hecho sus usuarios**.

El cliente busca la **portabilidad** quiere que el **software** se pueda acceder y ejecutar desde cualquier tipo de sistema operativo y desde cualquier dispositivo electrónico.

El cliente también quiere la **automatización de las tareas más rutinarias** como puede ser **pasar la información de un elemento del modelo a otro**, como por ejemplo pasar las facturas del diario de ventas al histórico de ventas de la empresa de una **forma automática y que el cliente no tenga que preocuparse de nada porque el software ya hace la labor por él**.

El cliente busca la **interoperabilidad que dos sistemas distintos se puedan intercambiar información entre ellos**. Como por ejemplo una aplicación de escritorio se pueda comunicar con un servidor.

Bajo estas **características** se cimentará la elaboración del proyecto.

Estudio de Viabilidad

Estudio del Mercado

Hoy en día los **Sistema de Planificación de Recursos Empresariales** su **comercialización está en auge porque ya no es tan necesario tener un alto poder adquisitivo para poder implementar uno**, claro está que con un gran apoyo financiero se puede implementar un **ERP** que cubra toda clase de funcionalidades, pero con un **presupuesto modesto** también se puede hacer grandes cosas. También hay que comentar que el **propio estado esta subvencionando a las pequeñas y medianas empresas** para que se **digitalicen y así puedan acceder a nuevas tecnologías**.

Todo lo dicho anteriormente **me beneficia** porque quiere decir que el **ERP** que **voy a implementar para gestionar y administrar una empresa tiene una alta demanda** por parte de **todo tipo de empresas**, por tanto, podría **intentar quedarme con una pequeña cuota de este mercado que está en auge**. Todo esto quiere decir que el **software** que **voy a distribuir es necesario y tiene hueco en el mercado** teniendo como **cliente objetivo a las PYMES** (Pequeñas y Medianas Empresas).

DAFO y CAME

He hecho uso de la herramienta **DAFO** (Debilidades, Amenazas, Fortalezas, Oportunidades) para hacer un análisis y estudio sobre la viabilidad que tiene la elaboración de este proyecto. La herramienta **DAFO** me brindara la oportunidad de diseñar una estrategia que saque el máximo partido del software que voy a implementar.

Nota: **DAFO** es una herramienta de estudio y análisis que permite al usuario analizar la realidad de su negocio, proyecto, marca o producto para poder saber la viabilidad que tiene la labor que vas a llevar a cabo.



Para aprovechar la **estrategia** que nos proporciona el **DAFO** tendremos que hacer uso de la herramienta **CAME** (Corregir, Afrontar, Mantener y Explorar), esta herramienta nos permitirá adaptar y corregir la estrategia que hemos diseñado en el **DAFO**. Básicamente la tarea que desempeñara el **CAME** es corregir las debilidades, afrontar las amenazas, explotar las oportunidades y mantener las fortalezas.

Nota: **CAME** es una herramienta de planificación que tiene como principal objetivo el estructurar los pasos a seguir con la organización. Permite adaptar las ideas establecidas en el **DAFO** para instalarlas en un plano de actividad real.



Viabilidad Técnica y Económica

Recursos de Hardware

Para la elaboración de este proyecto no se **necesita una gran potencia técnica** para el Servidor OODO ni para la aplicación de Escritorio.

El ERP OODO puede funcionar correctamente en un servidor o en otros dispositivos electrónicos con características técnicas bastante asequibles, porque solo necesita para funcionar un procesador dual Core, 2 GB RAM, pero lo más recomendable sería contar con **8 GB de RAM**, un **disco duro** con una capacidad mínima de **50 GB** y también necesitaría una **tarjeta de red** (para permitir las conexiones). En mi caso he instalado **ODOO** en una **máquina virtual**, esta máquina hará uso de los siguientes recursos del ordenador anfitrión, utilizará **dos núcleos Procesador I7-8750H 2,20GHz, 5GB de RAM y 150 GB** de espacio de disco duro y la **tarjeta de red INTEL 9560 NGW**. Y en lo que se refiere a la **aplicación de escritorio** cualquier **tipo de ordenador tanto actual como antiguo** podría ejecutar la **aplicación sin problemas**. Los gastos en **Hardware** serán bastantes **asequibles para la empresa** porque **no es necesario tener una gran potencia técnica**, aunque sí sería **recomendable tener un servidor o equipo que sobrepase y mejore los requerimientos mínimos** sobre todo por tema de **conexiones simultáneas de usuarios en el ERP**.

Por lo dicho anteriormente **sería recomendable hacer una buena inversión en Hardware** para la implementación del **ERP** porque esa inversión permitirá tener **un buen rendimiento del servidor y mejorará la eficiencia tanto del servidor como de los trabajadores**, **mejorará la eficiencia de los trabajadores** porque tendrán **un servidor bien implementado y estructurado** que permite la **conexión múltiple de usuarios**, **facilite la elaboración de la labor tanto conjuntamente como individualmente** y **evite cuelgues y fallos de rendimiento** por culpa de una **mala inversión en el Hardware**.

Recursos de Software

En lo que se refiere al **Software** he hecho **uso del Sistema de Planificación Empresarial OODO y del framework front-end Quasar**. Quasar es un **framework gratuito y de código abierto** por tanto no se requiere **ninguna inversión en ese Software**, pero en el caso del **Servidor OODO** que es un **ERP de código abierto** sí que se **podría hacer una inversión** porque **ODOO tiene ciertos módulos y funcionalidades que son de pago**, pero en la **elaboración de este proyecto** he utilizado la **versión comunitaria la cual es gratuita** y nos brinda una **serie de módulos básicos** que podemos utilizar **sin coste alguno**. La empresa por tanto no tendrá que hacer una **ninguna inversión de Software a no ser que necesite ciertos módulos y funcionalidades de pago** y también quiera **un soporte por parte de la empresa Odoo S.A** la creadora del **ERP**.

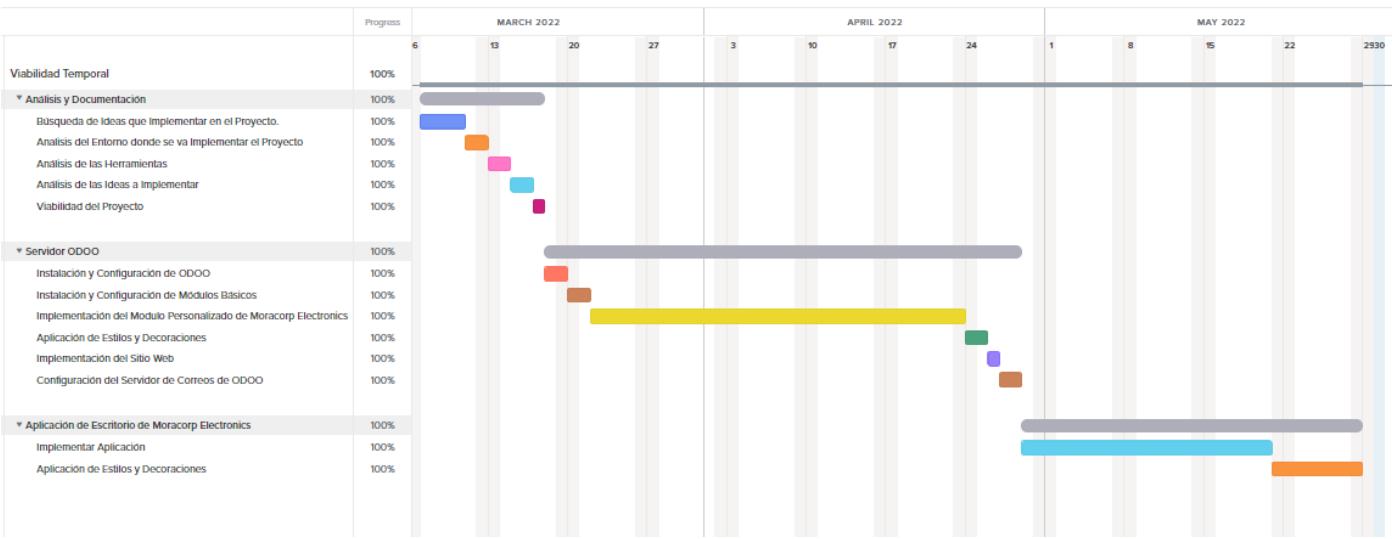
Recursos Humanos

Los **Recursos Humanos** utilizados para la elaboración de este proyecto han sido **Álvaro Mora López como programador y desarrollador del proyecto** y **Miguel Ángel Mora López como consejero y tester**.

Viabilidad Temporal

He creado un **diagrama de Gantt** para exponer la dedicación prevista para la elaboración del proyecto a lo largo de **un flujo de tiempo determinado**. He tenido de plazo para la elaboración del proyecto desde **1 de Marzo del 2022 hasta el 6 de Junio del 2022**. Al tener un **periodo de tiempo ya ajustado y determinado** he tenido que **planificar las etapas de desarrollo de los softwares que deseo implementar**.

Nota: El **diagrama de Gantt** es una herramienta **gráfica de planificación** que nos permite **remarcar o exponer el tiempo dedicado** que se **tiene previsto para la elaboración de una labor o actividad**.



Diseño

Diagrama Entidad Relación

En este apartado de la memoria explicare el **diseño conceptual de la base de datos**, por el cual se regirán las relaciones e interacciones de los modelos del módulo personalizado de la empresa **Moracorp Electronics** que voy a implementar.

La **tabla Producto / Stock** se relaciona con la **tabla de Proveedores** esta relación se fundamenta con el siguiente concepto:

- Diversos productos son suministrados a la empresa por un Proveedor.

La **tabla Compras** se relaciona con las tablas de **Productos y de Proveedores** estas relaciones se fundamentan con el siguiente concepto:

- La compañía realiza múltiples compras de un producto en particular a uno de sus proveedores.

La **tabla Ventas** se relaciona con las tablas de **Transacciones de Venta y de Clientes** estas relaciones se fundamentan con los siguientes conceptos:

- Múltiples ventas se realizan a un cliente en específico.
- En una sola venta se realizan múltiples transacciones de venta de un producto en particular.

La **tabla de Transacciones de Venta** se relaciona con la **tabla de Productos** esta relación se fundamenta con el siguiente concepto:

- Múltiples transacciones de venta se realizan de un producto en particular.

La **tabla de Pedidos** se relaciona con la **tabla de Ventas** esta relación se fundamenta con el siguiente concepto:

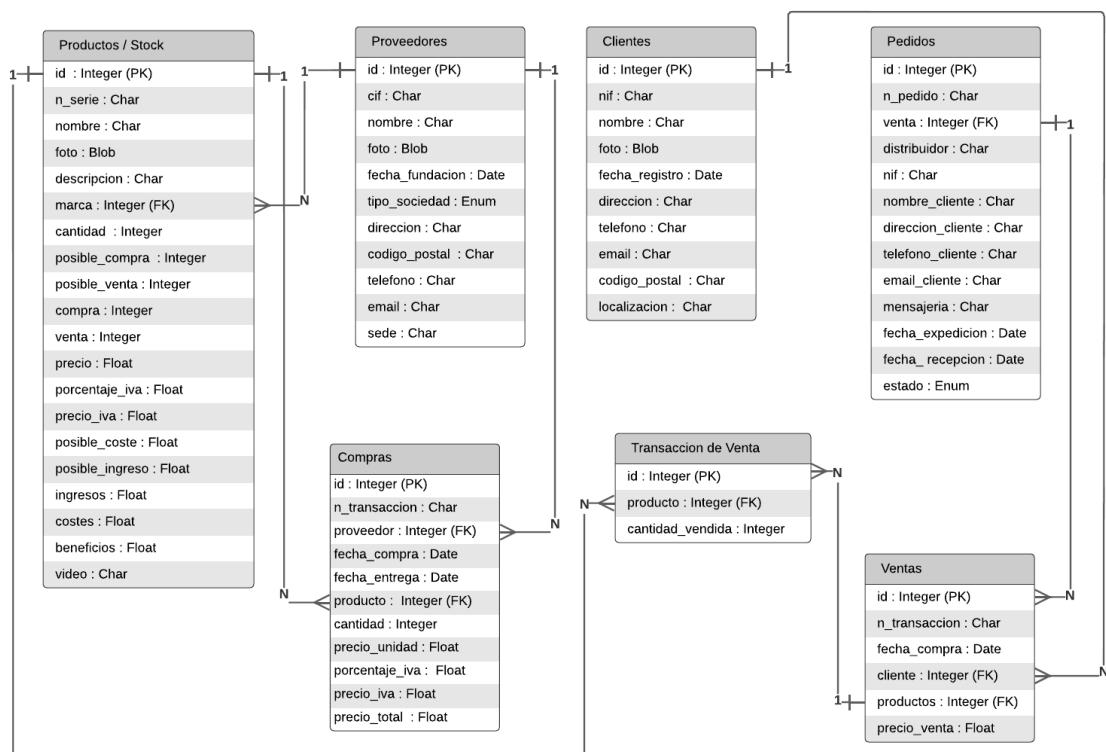
- En un único pedido se tramitan numerosas ventas.

Este es el **diseño de la base de datos que se implantará en el módulo personalizado para la empresa Moracorp Electronics**.



Diagrama Entidad Relacional

Moracorp Electronics



A lo largo de esta memoria se explicará con más detalle las relaciones y interacciones entre modelos del módulo hecho a la medida para la empresa Moracorp Electronics.

ODOO

Módulos Básicos

En el **ERP de OODOO** he instalado y configurado una serie de **módulos básicos** que nos brinda la **compañía Odoo S.A** que es la **creadora del ERP de OODOO**. Estos **módulos básicos** nos permitirán **administrar y gestionar las labores** de la empresa de una **forma organizada y eficiente**.

Módulo de Sitio Web

Este **módulo** nos permitirá **diseñar y desarrollar** nuestro propio **Sitio Web** dentro de nuestro **Servidor OODOO**. He **diseñado** un **Sitio Web** a la medida de la empresa **Moracorp Electronics**, este **Sitio Web** beneficiara a la empresa porque **mejorara la imagen de marca** permitiendo plasmar **todas las virtudes que representa la marca y todos los valores** que quiera **reflejar la empresa**. Este **Sitio Web** también beneficiara a la empresa dándole **una posición de marca** y un **reconocimiento tanto a nivel local como nacional**, otra **ventaja de crear** un **Sitio Web** es el acercamiento a **nuevos cliente y mercados**, este acercamiento desencadena otra ventaja la cual es un **mayor índice de ventas** al dirigirnos a un público más amplio. Un **Sitio Web** también nos beneficia porque nos **permitirá fidelizar a nuestros clientes** gracias a que **conoceremos sus gustos y sabremos qué es lo que quieren y que no quieren**. Básicamente **diseñar un Sitio Web** a la medida de la empresa, nos **permitirá crear un escaparate virtual** al que podrá **acceder cualquier persona en cualquier parte del mundo** para realizar una serie de compras e informarse de quienes somos que valores tiene nuestra empresa y cuáles son nuestras razones de ser.

The screenshot shows a professional website for 'Moracorp Electronics' built using the Odoo CMS module. The header features a blue-toned background image of a futuristic cityscape with glowing lights. The navigation bar includes links for 'Página', 'Personalizar', 'Promover', 'Inicio', 'Contáctenos', 'Administrator', and 'Contáctenos'. The main content area has a large banner with the company name 'Moracorp Electronics' and a subtext 'Bienvenidos a la Revolución de la venta tradicional y online de Productos Electronicos'. Below the banner are three cards: 'Pioneros' (with a laptop icon), 'Calidad' (with a hand interacting with a digital interface icon), and 'Garantías' (with a handshake icon). Further down is a section titled 'Quiénes Somos' featuring a golden shield logo with a bird emblem. At the bottom, there's a dark banner with the text 'Las Mejores Marcas estan Aquí' and various footer links and social media icons.

Módulo de Contactos

Este **módulo** nos permitirá **crear contactos dentro de nuestro ERP**. He configurado este **módulo** porque brindara a la **empresa** la posibilidad de **comunicarnos cómodamente y fácilmente con sus clientes y proveedores e incluso entre los mismos trabajadores de la empresa**. Este **módulo** permitirá tener una **comunicación directa y estrecha** con los **clientes y proveedores e incluso entre el personal que trabaja en la empresa de Moracorp Electronics**. Esta **comunicación beneficiara a la compañía para hacer marketing online** a los clientes enviado les ofertas, promociones, encuestas y **información de los nuevos servicios y productos**, y a su vez también le **beneficiara a la empresa para realizar tareas de atención al cliente y soporte sobre los productos y servicios que distribuyen**. También con **este módulo** la **compañía se comunicará con los proveedores**, esta **comunicación será muy beneficiosa** sobre todo por temas de **contratos, relaciones, garantías, soporte y pedidos de compra** entre la **empresa Moracorp Electronics y sus proveedores**. Y finalmente este **módulo beneficiara también la comunicación interna entre los trabajadores de la empresa Moracorp Electronics**. Este módulo como veremos en el **próximo apartado de la memoria** será crucial para la **comunicación del Servidor de Correos de nuestro ERP**.

The screenshot shows the Odoo Contacts module. At the top, there's a navigation bar with 'Contactos', 'Contactos', and 'Configuración'. On the right, it shows 'Administrator (Moracorp_Electronics)'. Below the navigation, there's a search bar and filter buttons ('Filtros', 'Agrupar por', 'Favoritos'). The main area displays two contacts in cards:

- Administrator**: Alcaya, España. Email: almotol7@gmail.com
- Moracorp Electronics**: Alcaya, España. Email: moracorp.electronics17@gmail.com

At the bottom right, there are pagination controls (1-2 / 2) and a grid view icon.

Servidor de Correos ODOO

Configurar los **servidores de correos de OODO** permitirá a la **empresa comunicarse eficientemente, cómodamente y fácilmente con sus clientes y proveedores e incluso entre los mismos trabajadores de la empresa**. Todo usuario del **ERP** podrá recibir y enviar información **sin necesidad de dirigirse a los gestores de correos electrónicos** que tenga dicho usuario. El **servidor de correo** nos permitirá hacer una **gestión ágil, flexible y eficiente de la comunicación corporativa**, configurarlo **beneficiará a la empresa en la Gestión de Proyectos, CRM (Gestión de Relaciones con el Cliente), Marketing automático, Gestión de Relaciones con los Proveedores y Facturación electrónica**.

Pasos Previos

El correo que vayamos a configurar como **correo saliente y entrante** tendrá que ser configurado previamente para que no de problemas con la vinculación con los servidores de correos de **ODOO**. Tendremos que habilitar **el acceso de las aplicaciones menos seguras dentro del gestor de correos del email** que vayamos a añadir a **ODOO** y también tendremos que habilitar el protocolo **IMAP**.

Implementación

El primer paso para configurar los servidores de correos será **dirigirnos a ajustes y activar el modo desarrollador**.

Herramientas de desarrollador

[Activar modo de desarrollador \(con activos\)](#)
[Activar el modo de desarrollador \(con activos de prueba\)](#)
[Desactivar modo de desarrollador](#)
[Cargar datos de prueba](#)

Dentro de la página de **ajustes** nos **dirigimos a la cabecera y seleccionamos el menú desplegable de la cabecera cuyo nombre es Técnico**. Al clicar sobre el menú se nos abrirá un menú desplegable con una **serie de opciones** en nuestro caso **seleccionaremos la opción servidores de correo saliente**.

Configuramos el **Correo Saliente** del usuario que deseemos en mi caso **configurare el correo del Administrador como correo saliente**. Para ello tendremos que **indicar la prioridad**, el servidor **SMTP** que vamos a usar variará según el gestor de correos que vayamos a utilizar en mi caso **utilizo como gestor de correos Gmail**, y también tendremos que asignar el puerto que hará uso del **protocolo SMTP** los más comunes para ello **son 25, 465, 587 y 2525**. Y ya para terminar la configuración del **Correo Saliente** tendremos que introducir nuestra **dirección de correo y contraseña y asignar la seguridad de la conexión**.

Nota: En caso de dar **fallo al probar la conexión con el correo introducido** en la configuración del **Servidor de Correo Saliente** deberá **comprobar la seguridad del correo y habilitar el acceso de las aplicaciones menos seguras**.

Descripción	Administrador	Prioridad
Activo	<input checked="" type="radio"/>	10
Información de la conexión		
Servidor SMTP	smtp.gmail.com	Puerto SMTP
Depurando	<input type="checkbox"/>	587
Seguridad y Autenticación		
Seguridad de la conexión	TLS (STARTTLS)	
Nombre de usuario	aimololi7@gmail.com	
Contraseña	*****	
<input type="button" value="Probar conexión"/>		

Una vez configurado el **Correo Saliente** volvemos a clicar sobre el menú desplegable **Técnico** y seleccionamos **la opción servidores de correo entrante**.

Ahora configuramos el **Correo Entrante** del usuario que anteriormente habíamos configurado su **Servidor de Correo Saliente**. Para ello tendremos que indicar el **tipo de servidor** en nuestro caso utilizaremos el **servidor que utiliza el protocolo IMAP**, introduciremos el **nombre del servidor** que **dependerá del gestor de correos electrónicos** que vayamos a usar en mi caso utilizar el de **Gmail**, y **asignamos el puerto** en este caso dejamos el que nos brinda por defecto. Lo siguiente que vamos a hacer es permitir el uso de los protocolos de seguridad **SSL/TLS** e introducir nuestra dirección de correo y contraseña.

Nota: En caso de **fallo** revise la **configuración del correo electrónico**, porque si no está habilitado el **protocolo IMAP** en el correo electrónico introducido en la configuración del **Servidor de Correo de Entrada** nos fallará y no podremos recibir los correos en nuestro servidor **ODOO**.

Nombre	Administrador	Tipo de servidor	Servidor IMAP
Última fecha de búsqueda	27/05/2022 10:44:06		

[Servidor e Inicio de sesión](#) [Avanzado](#)

Información del servidor

Nombre del servidor	imap.gmail.com
Puerto	993
SSL/TLS	<input checked="" type="checkbox"/>

Acciones a realizar en los correos entrantes

[Crear un nuevo registro](#)

Información de inicio de sesión

Nombre de usuario	almolol17@gmail.com
Contraseña	*****

Por último, para terminar con la configuración del **Correo Entrante** nos dirigimos a la **pestaña avanzado**, en ella **asignamos el nivel de prioridad del servidor y permitimos las opciones de guardar archivos adjuntos y mantener el original**.

Nombre	Administrador	Tipo de servidor	Servidor IMAP
Última fecha de búsqueda	27/05/2022 10:44:06		

[Servidor e Inicio de sesión](#) [Avanzado](#)

Opciones avanzadas

Prioridad del servidor	5
Mantener el original	<input checked="" type="checkbox"/>

Guardar archivos adjuntos	<input checked="" type="checkbox"/>
Activo	<input type="radio"/>

Una vez configurados los **Servidores de Correos Entrantes y Salientes** deberemos **ajustar el tiempo de sincronización** de ambos para ello **seleccionamos otra vez el menú Técnico y seleccionamos la opción Acciones Planificadas**.

Para ajustar el tiempo de **sincronización o actualización del Correo Entrante** deberemos dirigirnos a la opción de **Servicio de Fetchmail** de la **página de Acciones Planificadas** donde ajustaremos el **tiempo de sincronización o actualización del Servidor de Correos Entrantes** a un minuto.

Correo: servicio de Fetchmail

Modelo	Servidor de correo entrante	Disponible en el sitio web
Usuario del planificador	OdooBot	<input type="checkbox"/>
Ejecutar cada	1Minutos	
Activo	<input checked="" type="radio"/>	
Siguiente fecha de ejecución	27/05/2022 10:49:38	
Número de ejecuciones	-1	
Prioridad	5	
Repetir perdidos	<input type="checkbox"/>	

[Código Python](#) [Seguridad](#) [Ayuda](#)

```
1 _fetch_mails()
```

A continuación, **actualizaremos el tiempo de sincronización**, pero en este caso para el **Correo Saliente**, para ello tendremos que **dirigirnos a la opción de Gestor de colas de Email de la página de Acciones Planificadas** donde **ajustaremos el tiempo de sincronización o actualización del servidor de Correos Salientes a un minuto**.

Mail: Gestor de colas de email

Modelo	Correos salientes	Disponible en el sitio web
Usuario del planificador	OdooBot	
Ejecutar cada	1Minutos	
Activo	<input checked="" type="checkbox"/>	
Siguiente fecha de ejecución	27/05/2022 10:52:35	
Número de ejecuciones	-1	
Prioridad	5	
Repetir perdidos	<input type="checkbox"/>	

Código Python Seguridad Ayuda

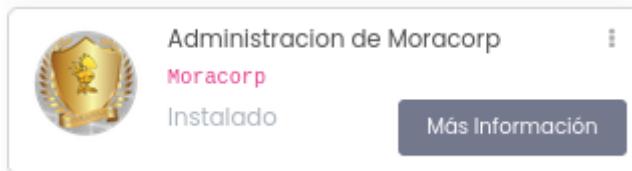
```
1 process_email_queue()
```

Por último, nos creamos un nuevo contacto en el módulo de Contactos para probar la correcta configuración de los Servidores de Entrada y Salida para el correo del usuario Administrador. En mi caso no me ha hecho falta crearme ningún contacto porque he utilizado el contacto que crea por defecto OODOO al instalarse, este contacto lo he adaptado con la información referente a la empresa Moracorp Electronics.

Como podemos observar en la imagen la configuración del servidor de Correo Entrante y Saliente para el correo del usuario Administrador ha sido todo un éxito porque nos permite recibir y enviar correos desde la cuenta de usuario Administrador de nuestro servidor OODOO.

Módulo de Administración Moracorp

Este módulo personalizado será uno de mis pilares fundamentales que sostendrá toda la estructura y razón de ser de este proyecto. Este módulo denominado Administración Moracorp permitirá gestionar y administrar las labores de la empresa Moracorp Electronics de una forma eficiente, rápida, sencilla y centralizada.



Este módulo se encargará principalmente de la gestión de los productos y del stock de los mismos, de las compras y ventas de productos, de la tramitación y administración de los pedidos y también se encargará de dar un buen soporte en las comunicaciones y relaciones con los clientes como con los distribuidores o proveedores. El módulo de Administración Moracorp también nos permitirá llevar a cabo una eficiente tramitación y administración de la documentación de la empresa haciendo uso de los informes.

Finalmente este módulo personalizado se comunicara con la aplicación de escritorio haciendo uso de una serie de APIS que se encuentran implementadas en unos Web Controllers, estas APIS nos permitirán poder observar la información de los modelos que componen el módulo y hacer toda clase de creaciones y modificaciones en los mismos, todo esto quiere decir que gracias a las APIS que hemos patentado en el Servidor OODOO podremos observar la información de los modelos desde la aplicación de escritorio, también podremos hacer creaciones y modificaciones desde la aplicación de escritorio que repercutirán directamente en los valores de los modelos del módulo de Administración Moracorp .

En este apartado de la memoria voy a explicar de una forma clara, sencilla y extensa paso a paso cómo funciona el módulo personalizado creado expresamente para la empresa de Moracorp Electronics, pero en función que se vayan repitiendo conceptos y acciones en los diversos modelos que componen el módulo iré abreviando y obviando explicaciones para no extenderme demasiado ,pero en lo que se refiere a funcionalidades que tiene los modelos y cómo utilizarlos estarán perfectamente desarrollados, explicados y documentados en esta memoria.

Modelo de Productos

Este **modelo se encargará de gestionar y administrar los productos** que va a **distribuir la empresa**. El modelo de productos brindara a la empresa **la posibilidad de registrar los productos que van a comercializar dentro del ERP de una forma sencilla e intuitiva**.

Productos de Moracorp Electrics									
Crear		Buscar...					Filtros		
ID	Nº de Serie	Nombre	Marca	Foto	Descripción		Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)
1	1	Teclado	Asus		Teclado RGB		12,00	21,00	14,52
2	2	Raton	Asus		Raton RGB		12,00	21,00	14,52
3	3	Portatil MSI	MSI		Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb		500,00	21,00	605,00
4	4	Auriculares	Asus		QUAD DAC™ ESS 9281		17,22	21,00	20,84

Relaciones del Modelo

El **modelo Productos se relaciona con el modelo de Proveedores por el campo marca** por lo tanto **debe existir un proveedor ya creado anteriormente que te venda ese producto o puedes crear dicho proveedor durante la creación del producto**.

Proveedores de Moracorp Electrics												
Crear		Buscar...					Filtros		Agrupar por		★ Favoritos	
ID	Código de Identificación Fiscal(CIF)	Nombre del Proveedor	Foto	Fecha de Fundación	Tipo Sociedad	Dirección del Proveedor	Código Postal	Teléfono	Email	Sede Central		
2	1	Asus		19/01/1989	Sociedad Anónima (S.A.)	C/ Metalurgia 38, Planta 1, 08038 Barcelona	08038	932 93 81 54	asus@gmail.com	Taipei,Taiwan		
3	2	MSI		17/01/1986	Sociedad Anónima (S.A.)	Avinguda Diagonal, 468, 08006 Barcelona	08006	931 845 542	msi@gmail.com	Taipei,Taiwan		

Restricciones y Utilidades del Modelo

Este **modelo cuenta con una restricción** la cual es que **el número de serie debe de ser único por lo tanto ningún producto nuevo registrado podrá repetir números de serie ya existentes**. En caso de repetir el número de serie el módulo impedirá el registro y le saltará un error por pantalla diciéndole que debe insertar un número de serie no repetido.

La **utilidad** que presenta este modelo es **un campo calculado que a partir del precio sin IVA y el porcentaje de IVA calcula automáticamente el precio del producto con el IVA incluido**.

Funcionamiento del Modelo

Una vez tenidas en cuenta las relaciones y restricciones comenzaremos con la creación del producto para ello clicamos sobre el botón crear, al clicar el botón se nos abrirá un formulario que deberemos llenar con toda la información relevante del producto.

Inmediatamente después de haber creado el producto se podrá observar en la ventana principal del modelo de productos, en esta ventana podremos ver todos los productos que registremos.

ID	Nº de Serie	Nombre	Marca	Foto	Descripción	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)
1	1	Tecclado	Asus		Tecclado RGB	12,00	21,00	14,52
2	2	Raton	Asus		Raton RGB	12,00	21,00	14,52
3	3	Portatil MSI	MSI		Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb	500,00	21,00	605,00
4	4	Auriculares	Asus		QUAD DAC™ ESS 9281	17,22	21,00	20,84
81	5	ASUS TUF VG27AQ	Asus		Monitor 4K	350,00	21,00	423,50

Cada vez que creamos un producto se actualizará el modelo stock y añadirá dicho producto al stock, pero con todos sus valores a cero porque todavía no se ha hecho ninguna compra y por tanto no se puede vender porque todavía no tenemos stock, y como no hemos hecho aún ninguna compra y venta no tenemos ni costes, gastos ni beneficios.

Stock de Moracorp Electrics								
Crear		Filtros Agrupar por Favoritos						
ID	Nombre	Cantidad en stock	Compra	Venta	Ingresos	Costes	Beneficios	
1	Tecclado	0	7	7	84,00	61,41	22,59	
2	Raton	0	2	2	24,00	19,98	4,04	
3	Portatil MSI	0	2	2	1.000,00	968,00	32,00	
4	Auriculares	0	2	2	34,44	24,20	10,24	
81	ASUS TUF VG27AQ	0	0	0	0,00	0,00	0,00	

Si quisiéramos **editar un producto ya creado** sería tan sencillo como **clicar sobre el producto que deseamos modificar**, **inmediatamente después de haber clicado el producto se nos abrirá el formulario** con los datos referentes al producto seleccionado, pero para **poder editar dicho formulario** tendremos que darle al **botón de editar** y ya nos **permitirá modificar los valores del producto**.

Si quisiéramos **eliminar un producto** **hacemos clic sobre la casilla seleccionable** (si quisiéramos borrar más de un producto tendríamos que clicar sobre las casillas de los demás productos que deseamos borrar) **de la fila del producto** y posteriormente **clicamos sobre el botón acción** y se nos **abrirá un menú desplegable** **seleccionamos la opción suprimir y nos borrará el producto**. Si quisiéramos **borrar todos los productos** sería hacer lo mismo que antes, pero en vez de seleccionar la casilla que hay en la fila de todos los productos tendremos que **seleccionar la casilla que está junto a las cabeceras de los campos de los productos** esta casilla seleccionara la casilla de todos los demás productos y solo tendremos que clicar sobre el **botón acción** y **seleccionar la opción suprimir**.

Tanto si **borramos** cómo **modificamos** el módulo stock se **actualizará con los cambios que se hayan hecho**.

Vistas del Modelo

En este mismo **modelo** se ha implementado una serie de **vistas** que facilitan la observación de la información según las necesidades que tenga el usuario. Para cambiar de vista tendremos que clicar sobre estos botones (El primer botón lleva a la **vista Tree**, el segundo lleva a la **vista Pivot** y el último lleva a la **vista Kanban**).



La **vista que tiene por defecto el modelo es una vista de tipo Tree** (lista) donde se puede **observar la información referente a los productos en formato lista**, esta vista nos permitirá ver la información de una forma más agrupada y detallada, pero tener tanta información en pantalla puede ser algo abrumador y esto implique que **cuesta encontrar la información, por ello he implementado otras dos vistas sobre este modelo como son la Kanban y Pivot**.

ID	Nº de Serie	Nombre	Marca	Foto	Descripción	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Pùblico(PVP)
1	1	Teclado	Asus		Teclado RGB	12,00	21,00	14,52
2	2	Raton	Asus		Raton RGB	12,00	21,00	14,52
3	3	Portatil MSI	MSI		Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb	500,00	21,00	605,00
4	4	Auriculares	Asus		QUAD DAC™ ESS 9281	17,22	21,00	20,84
5	5	ASUS TUF VG27AQ	Asus		Monitor 4k	350,00	21,00	423,50

La **vista Pivot** permite observar de una forma más sencilla la información referente al aspecto más económico y contable de los productos. Básicamente esta **vista nos permite ver el precio sin IVA, el porcentaje de IVA y el precio con IVA de los productos**. Esta vista también nos brinda la **posibilidad de poder agrupar los productos por una serie de filtros determinados por los campos en el caso del modelo de productos estoy agrupando los valores de los productos por la marca**, estos filtros son muy útiles porque hacen que la obtención de la información sea más rápida y eficiente.

	- Total	+ Asus	+ MSI	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)
- Total	391,22	84,00		473,38	500,00	21,00	605,00	891,22	105,00	1.078,38		
+ ASUS TUF VG27AQ	350,00	21,00		423,50			350,00	21,00	423,50			
+ Auriculares	17,22	21,00		20,84			17,22	21,00	20,84			
+ Portatil MSI				500,00	21,00	605,00	600,00	21,00	605,00			
+ Raton	12,00	21,00		14,52			12,00	21,00	14,52			
+ Teclado	12,00	21,00		14,52			12,00	21,00	14,52			

La última vista que he aplicado sobre el modelo productos es la **Kanban** esta vista nos permitirá ver la información de los productos de una forma más gráfica e intuitiva. Es una vista más amigable con el usuario, pero no es tan completa de información como es la **Tree** tal y como yo he implementado las estructuras de las vistas.



Informe del Modelo

Finalmente en este modelo también he implementado un informe de productos donde se podrá observar todos las características y valores referentes a los productos que tenemos registrados en nuestra empresa ,para acceder al informe de productos deberemos seleccionar las casillas de cada producto registrado que deseamos mostrar en el informe , una vez seleccionadas las casillas deberemos clicar sobre el botón imprimir y seleccionar la opción Stock de Productos para poder acceder al informe de productos. He implementado este informe para observar de una forma rápida, sencilla y detallada toda la información más relevante de los productos que tenemos registrados.



Moracorp Electronics
Plaza España
46960 Aldaya
España

Stock del Producto

Identificador	Numero de Serie	Marca	Nombre del Producto	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)	Cantidad	Compra	Venta	Ingresos	Costes	Beneficios
81	5	Asus	ASUS TUF VG27AQ	350,0	21,0	423,5	0	0	0	0,0	0,0	0,0

Modelo de Compras

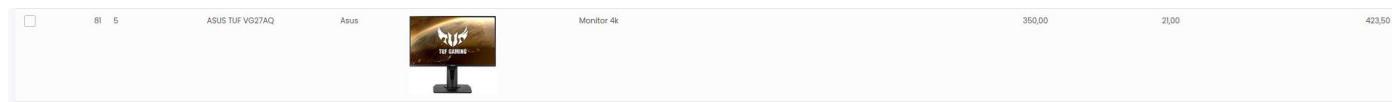
Este modelo se encargará de gestionar y administrar las compras de cada producto a cada uno de los proveedores que tiene la empresa. El modelo de Compras brindara a la empresa la posibilidad de registrar las compras que se han hecho a los distintos proveedores.

Para dirigirnos al modelo de compras deberemos seleccionar la opción Compras en el menú que se encuentra en la cabecera de la página web, si quisiéramos acceder otra vez a Productos haríamos lo mismo, pero seleccionando la opción Productos de la cabecera de la página web.

ID	Nº de Transaccion	Nombre del Proveedor	Fecha de Compra	Fecha de Entrega	Producto	Precio por Unidad sin IVA	Porcentaje de IVA	Precio por Unidad con IVA	Cantidad	Precio Total de la Compra
3	1	Asus	08/05/2022	10/05/2022	Teclado	7,25	21,00	8,77	6	52,63
4	2	Asus	10/05/2022	14/05/2022	Raton	8,25	21,00	9,98	2	19,96
5	3	MSI	09/05/2022	21/05/2022	Portatil MSI	400,00	21,00	484,00	2	968,00
6	4	Asus	02/05/2022	27/05/2022	Teclado	7,25	21,00	8,77	1	8,77
7	5	Asus	09/05/2022	12/05/2022	Auriculares	10,00	21,00	12,10	2	24,20

Relaciones del Modelo

El modelo Compras se relaciona con el modelo de productos por el campo producto por lo tanto tendríamos que tener un producto ya creado anteriormente para poder vincularlo con la compra o también podemos crear el producto mientras creamos la compra.



Y a su vez este modelo se relaciona con el modelo de proveedores por el campo nombre del proveedor por lo tanto debe existir un proveedor ya creado anteriormente que te venda ese producto o puedes crear dicho proveedor durante la creación de la compra.

ID	Código de Identificación Fiscal(Cif)	Nombre del Proveedor	Foto	Fecha de Fundación	Tipo Sociedad	Dirección del Proveedor	Código Postal	Teléfono	Email	Sede Central
2	1	Asus		19/01/1989	Sociedad Anónima (S.A.)	C/ Metalurgia 38, Planta 1, 08038 Barcelona	08038	932 93 81 54	asus@gmail.com	Taipei,Taiwan

Restricciones y Utilidades del Modelo

Este modelo cuenta con tres restricciones, la primera restricción es que el número de transacción no se puede repetir, la segunda restricción es que la fecha de compra no puede ser posterior a la fecha de entrega.

Y la tercera y última restricción es que la cantidad de productos comprados a nuestros proveedores no puede ser menor o igual a cero. Si incumplimos algunas de estas restricciones no se podrá realizar el registro de la compra y deberemos rectificar los errores que nos hayan ocurrido para poder tramitar el registro de compra.

El modelo de Compras cuenta con dos utilidades, la primera utilidad es que cuenta con un campo calculado que calcula el IVA del producto que vamos a comprar a partir del precio sin IVA y el porcentaje de IVA del producto.

La segunda utilidad es que cuenta con otro campo calculado que calcula automáticamente el precio total de la compra a partir del precio con IVA y la cantidad de producto comprado.

Funcionamiento del Modelo

Una vez **tenidas en cuenta las relaciones y restricciones comenzaremos con la creación de la compra** para ello clicamos sobre el botón crear, al clicar el botón se nos abrirá un formulario que deberemos llenar con toda la información relevante a la compra.

Nº de Transaccion	6	Nombre del Proveedor	Asus
Fecha de Compra	09/05/2022	Fecha de Entrega	13/05/2022
Producto	ASUS TUF VG27AQ	Precio por Unidad sin IVA	245,99
Porcentaje de IVA	21,00	Cantidad	2

<input type="checkbox"/>	126	6	Asus	09/05/2022	13/05/2022	ASUS TUF VG27AQ	245,99	21,00	297,65	2	595,30
--------------------------	-----	---	------	------------	------------	-----------------	--------	-------	--------	---	--------

Justo después de **asignar la compra de un producto y la cantidad comprada del mismo se actualizará automáticamente la cantidad en stock, la cantidad comprada, los costes y los beneficios en el modelo de Stock.**

<input type="checkbox"/>	ID	Nombre	Cantidad en Stock	Compra	Venta	Ingresos	Costes	Beneficios
<input type="checkbox"/>	1	Teclado	0	7	7	84,00	61,41	22,59
<input type="checkbox"/>	2	Raton	0	2	2	24,00	19,96	4,04
<input type="checkbox"/>	3	Portatil MSI	0	2	2	1.000,00	968,00	32,00
<input type="checkbox"/>	4	Auriculares	0	2	2	34,44	24,20	10,24
<input type="checkbox"/>	81	ASUS TUF VG27AQ	2	2	0	0,00	595,30	-595,30

Si editamos la compra que hemos hecho, **automáticamente en el modelo de Stock se actualizarán los valores que hayamos modificado.**

Si borramos la compra también se **actualizará automáticamente los valores referentes al producto comprado en el modelo Stock.**

Vistas del Modelo

En este **modelo implementaremos dos vistas la Tree y la Pivot.**

La **vista Tree** nos permitirá **ver la información más agrupada y detallada de las compras que hayamos hecho.**

<input type="checkbox"/>	ID	Nº de Transaccion	Nombre del Proveedor	Fecha de Compra	Fecha de Entrega	Producto	Precio por Unidad sin IVA	Porcentaje de IVA	Precio por Unidad con IVA	Cantidad	Precio Total de la Compra
<input type="checkbox"/>	3	1	Asus	08/05/2022	10/05/2022	Teclado	7,25	21,00	8,77	6	52,63
<input type="checkbox"/>	4	2	Asus	10/05/2022	14/05/2022	Raton	8,25	21,00	9,98	2	19,96
<input type="checkbox"/>	5	3	MSI	09/05/2022	21/05/2022	Portatil MSI	400,00	21,00	484,00	2	968,00
<input type="checkbox"/>	6	4	Asus	02/05/2022	27/05/2022	Teclado	7,25	21,00	8,77	1	8,77
<input type="checkbox"/>	7	5	Asus	09/05/2022	12/05/2022	Auriculares	10,00	21,00	12,10	2	24,20
<input type="checkbox"/>	126	6	Asus	09/05/2022	13/05/2022	ASUS TUF VG27AQ	245,99	21,00	297,65	2	595,30

La **vista Pivot** nos permitirá extraer más fácilmente la información referente a la contabilidad y gestión económica de los productos comprados.

	Total		MSI	
	Asus		mayo	
	mayo	2022	mayo	2022
Precio por Unidad sin IVA	Precio por Unidad con IVA	Precio por Unidad con IVA	Precio por Unidad sin IVA	Precio por Unidad con IVA
Cantidad	Precio Total de la Compra	Precio por Unidad sin IVA	Precio por Unidad con IVA	Precio por Unidad con IVA
		Cantidad	Precio Total de la Compra	
- Total	278,74	105,00	337,28	13
+ Teclado	14,50	42,00	17,55	7
+ Raton	8,25	21,00	9,98	2
+ Portatil MSI				
+ Auriculares	10,00	21,00	12,10	2
+ ASUS TUF VG27AQ	245,99	21,00	297,65	2
			595,30	

Informe del Modelo

Finalmente, en este modelo también he implementado un informe de compras donde se podrá observar las facturas de compras de los productos, para acceder al informe de facturas de compras deberemos seleccionar las casillas de cada fila de compra que deseamos mostrar en el informe, una vez seleccionadas las casillas deberemos clicar sobre el botón imprimir y seleccionar la opción Factura de Compra para poder acceder al informe de compra. Este informe lo podremos ver en vista previa sin descargarlo o también podríamos descargarlo para imprimirla y archivarla de forma física o descargarla y almacenarla en el disco duro del ordenador.

He implementado este informe para facilitar la gestión financiera de la empresa porque gracias a él se puede gestionar más fácilmente los historiales de facturas de compras y los diarios de contabilidad.



Moracorp Electronics
Plaza España
46960 Aldaya
España

Factura de Compra

Datos de la Transaccion:

Numero de Transaccion: 6

Fecha de Compra: 09/05/2022

Datos del Proveedor:

Fecha de Entrega: 13/05/2022

Nombre del Proveedor: Asus

Direccion del Proveedor: C/ Metalurgia 38, Planta 1, 08038 Barcelona

Ciudad: Taipei,Taiwan CP: 08038

Telefono: 932 93 81 54

Email: asus@gmail.com

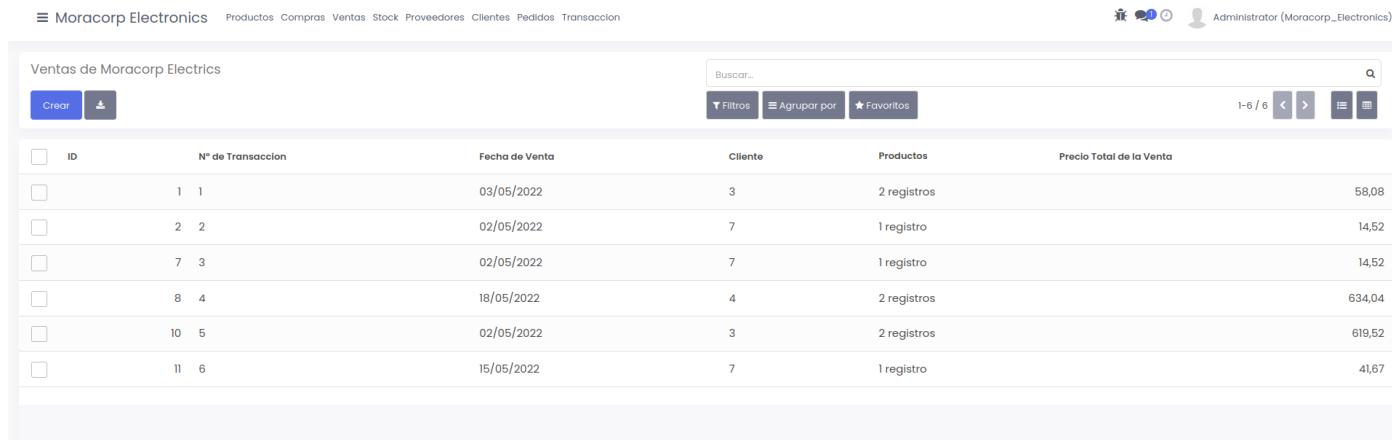
Producto	Cantidad	Precio Unidad sin IVA
ASUS TUF VG27AQ	2	245,99
Porcentaje de IVA: 21,0 %		
Precio Unidad con IVA: 297,6479 €		
Total de Compra: 595,2958 €		

6921002890 moracorp.electronics17@gmail.com

Página: 1/1

Modelo de Ventas

Este modelo se encargará de la gestión y la administración de las ventas de los productos que ha distribuido nuestra empresa. El modelo de Ventas permitirá a la empresa registrar las ventas de los productos que han vendido a sus clientes.



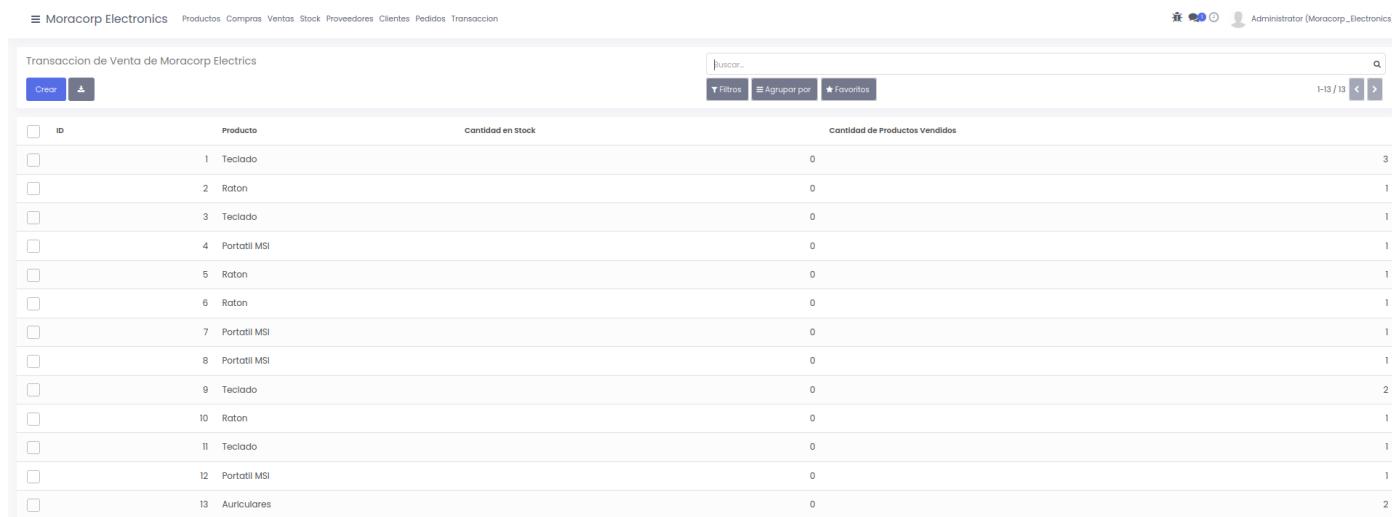
A screenshot of a web-based application interface for managing sales transactions. The top navigation bar includes links for Products, Purchases, Sales, Stock, Suppliers, Customers, Orders, and Transactions. The user is identified as 'Administrator (Moracorp_Electronics)'.

The main content area displays a table titled 'Ventas de Moracorp Electrics'. The table has columns for ID, Transaction Number, Date of Sale, Client, Products, and Total Sale Price. There are 6 rows of data, each showing a unique transaction with its details.

ID	Nº de Transaccion	Fecha de Venta	Cliente	Productos	Precio Total de la Venta
1	1	03/05/2022	3	2 registros	58,08
2	2	02/05/2022	7	1 registro	14,52
7	3	02/05/2022	7	1 registro	14,52
8	4	18/05/2022	4	2 registros	634,04
10	5	02/05/2022	3	2 registros	619,52
11	6	15/05/2022	7	1 registro	41,67

Relaciones del Modelo de Ventas

El modelo de Ventas se relaciona con el modelo de Transacciones de venta por el campo **productos**, por lo tanto, cada vez que añadamos un producto a la venta deberemos crear una transacción de venta, donde deberemos indicar qué producto queremos vender y la cantidad que queremos vender siempre y cuando tengamos el stock necesario de ese producto, porque si no tenemos el stock necesario y no podemos soportar la transacción de la venta dará un error y no nos permitirá llevarla a cabo. Y si no tenemos el stock necesario para poder vender tendremos que paralizar o postergar la venta hasta que haya stock. Y para que haya stock tendríamos que dirigirnos al modelo de Compras y realizar unas compras a nuestros proveedores.



A screenshot of a web-based application interface for managing sales transactions. The top navigation bar includes links for Products, Purchases, Sales, Stock, Suppliers, Customers, Orders, and Transactions. The user is identified as 'Administrator (Moracorp_Electronics)'.

The main content area displays a table titled 'Transaccion de Venta de Moracorp Electrics'. The table has columns for ID, Product, Stock Quantity, and Sold Product Quantity. There are 13 rows of data, each showing a product and its current stock level.

ID	Producto	Cantidad en Stock	Cantidad de Productos Vendidos
1	Teclado	0	3
2	Raton	0	1
3	Teclado	0	1
4	Portatil MSI	0	1
5	Raton	0	1
6	Raton	0	1
7	Portatil MSI	0	1
8	Portatil MSI	0	1
9	Teclado	0	2
10	Raton	0	1
11	Teclado	0	1
12	Portatil MSI	0	1
13	Auriculares	0	2

Este modelo también se relaciona con el modelo de clientes por el campo **cliente**, por consiguiente, tendríamos que haber creado un cliente previamente para relacionarlo con la venta o durante la creación de la venta podemos registrar un nuevo cliente y vincularlo a la nueva venta que queremos hacer.

Clientes de Moracorp Electrics									
<input type="checkbox"/>	ID	Nombre de identificación Fiscal(Ver)	Foto	Fecha de Registro	Direccion	Teléfono	Email	Código Postal	Localización
<input type="checkbox"/>	3	1		06/05/2022	Plaza España	692100284	all7mora@gmail.com	46960	Aldaya
<input type="checkbox"/>	4	3		02/05/2022	Calle Atocha	(46) 546 - 564 - 56...	miki@gmail.com	46960	Aldaya
<input type="checkbox"/>	7	4		02/05/2022	Calle Ademuz	692100297	angeles@gmail.com	46001	Valencia

Restricciones y Utilidades del Modelo

Este **modelo** tendrá tres restricciones fundamentales, la primera restricción es que el número de transacción no se puede repetir, la segunda restricción es que debe existir un stock de los productos para poder realizar la venta y la tercera restricción que tiene este **modelo** es que al menos se debe vender un producto.

La utilidad que presenta este **modelo** es que tiene un campo calculado que en función a que productos vendamos y la cantidad que hemos vendido de los mismos se sacará automáticamente el precio total de la venta.

Funcionamiento del Modelo

Una vez tenidas en cuenta las relaciones y restricciones comenzaremos con la creación de la venta para ello clicamos sobre el botón crear, al clicar el botón se nos abrirá un formulario que deberemos llenar con toda la información relevante a la venta.

Nº de Transaccion	7	Fecha de Venta	23/05/2022								
Cliente	3	Productos	<table border="1"> <thead> <tr> <th>ID</th> <th>Producto</th> <th>Cantidad en Stock</th> <th>Cantidad de Productos Vendidos</th> </tr> </thead> <tbody> <tr> <td></td> <td>ASUS TUF VG27AQ</td> <td>2</td> <td>1</td> </tr> </tbody> </table>	ID	Producto	Cantidad en Stock	Cantidad de Productos Vendidos		ASUS TUF VG27AQ	2	1
ID	Producto	Cantidad en Stock	Cantidad de Productos Vendidos								
	ASUS TUF VG27AQ	2	1								

Añadir: Productos

Producto	ASUS TUF VG27AQ	Cantidad en Stock	2
Cantidad de Productos Vendidos	1		

Nº de Transacción	7	Fecha de Venta	23/05/2022
Cliente	3	Productos	
	ID	Producto	Cantidad en Stock
	48	ASUS TUF VG27AQ	1

Justo después de realizar la venta de los productos y las cantidades vendidas de los mismos se actualizará automáticamente en el modelo de Stock la cantidad en stock, las ventas, los ingresos y beneficios.

<input type="checkbox"/>	ID	Nombre	Cantidad en Stock	Compra	Venta	Ingresos	Costes	Beneficios
<input type="checkbox"/>	1	Teclado	0	7	7	84,00	61,41	22,59
<input type="checkbox"/>	2	Raton	0	2	2	24,00	19,96	4,04
<input type="checkbox"/>	3	Portatil MSI	0	2	2	1.000,00	968,00	32,00
<input type="checkbox"/>	4	Auriculares	0	2	2	34,44	24,20	10,24
<input type="checkbox"/>	81	ASUS TUF VG27AQ	1	2	1	350,00	595,30	-245,30

Si editamos la venta que hemos hecho, automáticamente en el modelo de Stock se actualizarán los valores que hayamos modificado.

Si borramos la venta también se actualizará automáticamente los valores referentes a los productos vendidos en el modelo Stock.

Vistas del Modelo

En este modelo implementaremos dos vistas la Tree y la Pivot.

La vista Tree nos permitirá ver la información más agrupada y detallada de las ventas que hayamos hecho.

<input type="checkbox"/>	ID	Nº de Transacción	Fecha de Venta	Cliente	Productos	Precio Total de la Venta
<input type="checkbox"/>	1	1	03/05/2022	3	2 registros	58,08
<input type="checkbox"/>	2	2	02/05/2022	7	1 registro	14,52
<input type="checkbox"/>	7	3	02/05/2022	7	1 registro	14,52
<input type="checkbox"/>	8	4	18/05/2022	4	2 registros	634,04
<input type="checkbox"/>	10	5	02/05/2022	3	2 registros	619,52
<input type="checkbox"/>	11	6	15/05/2022	7	1 registro	41,67
<input type="checkbox"/>	95	7	23/05/2022	3	1 registro	423,50

La vista Pivot nos permitirá extraer más fácilmente la información referente a la contabilidad y gestión económica de los productos vendidos.

— Total				
— mayo 2022				
+ 3	+ 4	+ 7		
Precio Total de la Venta				
— Total	1.101,10	634,04	70,71	1.805,85
+ 1	58,08			58,08
+ 2			14,52	14,52
+ 3			14,52	14,52
+ 4		634,04		634,04
+ 5	619,52			619,52
+ 6			41,67	41,67
+ 7	423,50			423,50

Informe del Modelo

Finalmente, en este **modelo** también he implementado un informe de ventas donde se podrá observar las facturas de ventas de los productos, para acceder al informe de facturas de ventas deberemos seleccionar las casillas de cada fila de venta que deseamos mostrar en el informe, una vez seleccionadas las casillas deberemos clicar sobre el botón imprimir y seleccionar la opción Factura de Venta para poder acceder al informe de venta.

He implementado este informe para facilitar la gestión financiera de la empresa porque gracias a él se puede gestionar más fácilmente los historiales de facturas de ventas y los diarios de contabilidad.



Moracorp Electronics
Plaza España
46960 Aldaya
España

Factura de Venta

Datos de la Transaccion:

Numero de Transaccion: 7

Fecha de Compra: 23/05/2022

NIF: 1

Nombre: Alvaro

Direccion: Plaza España

Ciudad: Aldaya
CP: 46960

Teléfono: 692100284

Email: al17mora@gmail.com

Nombre del Producto	Cantidad en Stock	Cantidad Vendida
ASUS TUF VG27AQ	1	1

Total de Compra: 423,5 €

6921002890 moracorp.electronics17@gmail.com

Página: 1/1

Modelo Stock

Este modelo se encargará de la gestión y administración del inventariado y de la viabilidad de los productos. El modelo Stock permitirá saber a la empresa el stock almacenado que tiene de cada producto y además analizar y observar los beneficios y las pérdidas que produce cada producto.

Stock de Moracorp Electrics								
<input type="checkbox"/>	ID	Nombre	Cantidad en Stock	Compra	Venta	Ingresos	Costes	Beneficios
<input type="checkbox"/>	1	Teclado	0	7	7	84,00	61,41	22,59
<input type="checkbox"/>	2	Raton	0	2	2	24,00	19,96	4,04
<input type="checkbox"/>	3	Portatil MSI	0	2	2	1.000,00	968,00	32,00
<input type="checkbox"/>	4	Auriculares	0	2	2	34,44	24,20	10,24
<input type="checkbox"/>	81	ASUS TUF VG27AQ	1	2	1	350,00	595,30	-245,30

Relaciones del Modelo

El modelo Stock no es un modelo como tal porque no es un modelo independiente que se relaciona con otros modelos, el modelo Stock forma parte del modelo de Productos por tanto el modelo Stock y el modelo Productos son el mismo modelo, pero con diferentes vistas que muestran información diferente, pero en lo que se refiere al funcionamiento interno de ambos modelos es el mismo. Por lo tanto, el modelo Stock tiene las mismas relaciones, restricciones y utilidades que tiene el modelo de productos. Aunque tenga las mismas utilidades hay ciertas utilidades que están ocultas en el modelo de Productos y solo se muestran en el modelo de Stock.

Restricciones y Utilidades del Modelo

El modelo Stock tiene la misma restricción que el modelo Productos, la cual es que el número de serie del producto inventariado no se puede repetir.

El modelo Stock cuenta con seis utilidades, la primera utilidad es que cuenta con el campo calculado stock que determina automáticamente la cantidad de productos que hay en stock teniendo en cuenta la compras y ventas del producto.

La segunda utilidad es un campo calculado de compras que se actualiza automáticamente en función a las compras que se hagan de un producto en particular.

La tercera utilidad es un campo calculado de ventas que se actualiza automáticamente en función a las ventas que se hagan de un producto en particular.

La cuarta utilidad es un campo calculado de los ingresos que se determina en función de los precios de venta sin IVA de un producto y la cantidad vendida del mismo.

La quinta utilidad es un campo calculado de los costes que se determina en función de los precios de compra con IVA de un producto y la cantidad comprada del mismo.

Y la sexta y última utilidad es un campo calculado que determina los beneficios en función de los costes y los ingresos que nos aportan un producto en particular.

Funcionamiento del Modelo

En este modelo se podrá crear, eliminar y modificar, pero todo ello repercutirá al modelo **Productos**, esto significa que el modelo **Stock** tiene todas las funcionalidades que tiene el modelo **Producto**, pero con el añadido de la gestión del inventariado y de la viabilidad de los productos. Básicamente esto quiere decir que este modelo es un modelo más de análisis de información porque sus valores dependen de las compras y las ventas de los productos, aunque también incluye una faceta más de acción como es la administración de productos como hace el modelo de **Productos**.

Vistas del Modelo

En este modelo he aplicado tres vistas la Tree, la Pivot y la Graph.

La vista Tree nos permitirá ver la información más agrupada y detallada de los productos que tenemos inventariados y de la viabilidad de estos.

ID	Nombre	Cantidad en Stock	Compra	Venta	Ingresos	Costes	Beneficios
1	Teclado	0	7	7	84,00	61,41	22,59
2	Raton	0	2	2	24,00	19,96	4,04
3	Portatil MSI	0	2	2	1.000,00	968,00	32,00
4	Auriculares	0	2	2	34,44	24,20	10,24
81	ASUS TUF VG27AQ	1	2	1	350,00	595,30	-245,30

La vista Pivot nos permitirá extraer más fácilmente la información referente a la contabilidad y gestión económica de los productos que tenemos inventariados.

	Total	Asus	MSI
Precio sin IVA	391,22	84,00	
Porcentaje de IVA			
Precio de Venta al Público(PVP)	473,38		
Cantidad en Stock	1	13	12
Compra	492,44	700,87	-208,43
Venta	500,00	21,00	21,00
Ingresos	595,30	-245,30	
Costes			
Beneficios	21,00	605,00	32,00
Total	391,22	84,00	
+ ASUS TUF VG27AQ	350,00	21,00	
+ Auriculares	17,22	21,00	
+ Portatil MSI			
+ Raton	12,00	21,00	
+ Teclado	12,00	21,00	

La vista Graph nos permitirá ver de una forma más gráfica la viabilidad que tiene cada producto porque nos mostrará mediante todo tipo de gráficos (gráfico de barras, gráfico de líneas y gráfico circular) los beneficios y pérdidas que producen cada producto que tenemos registrado en nuestro ERP.

Gráfico de Líneas:

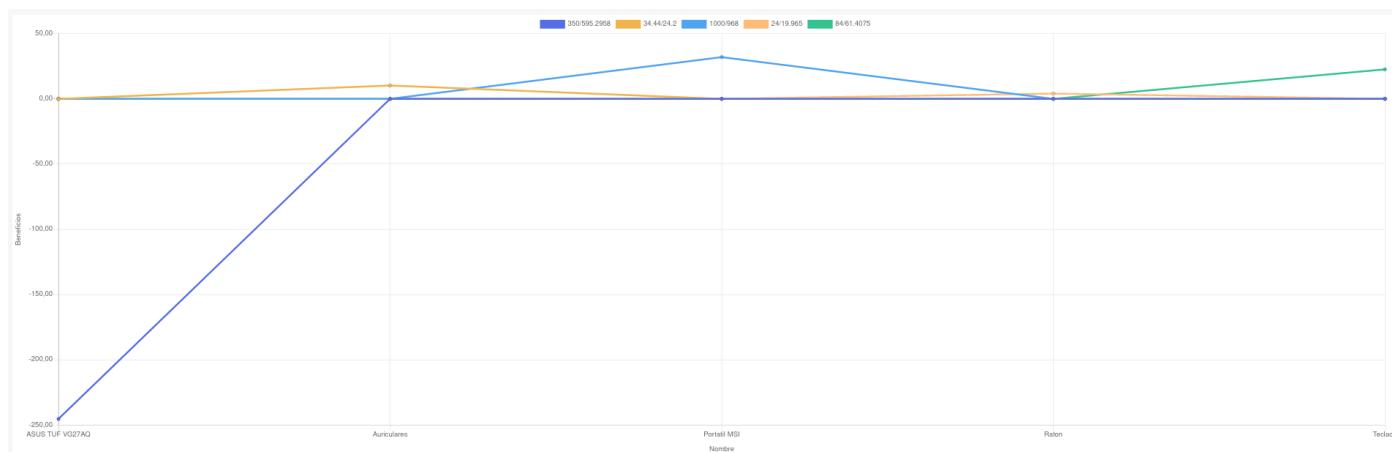


Gráfico de Barras:

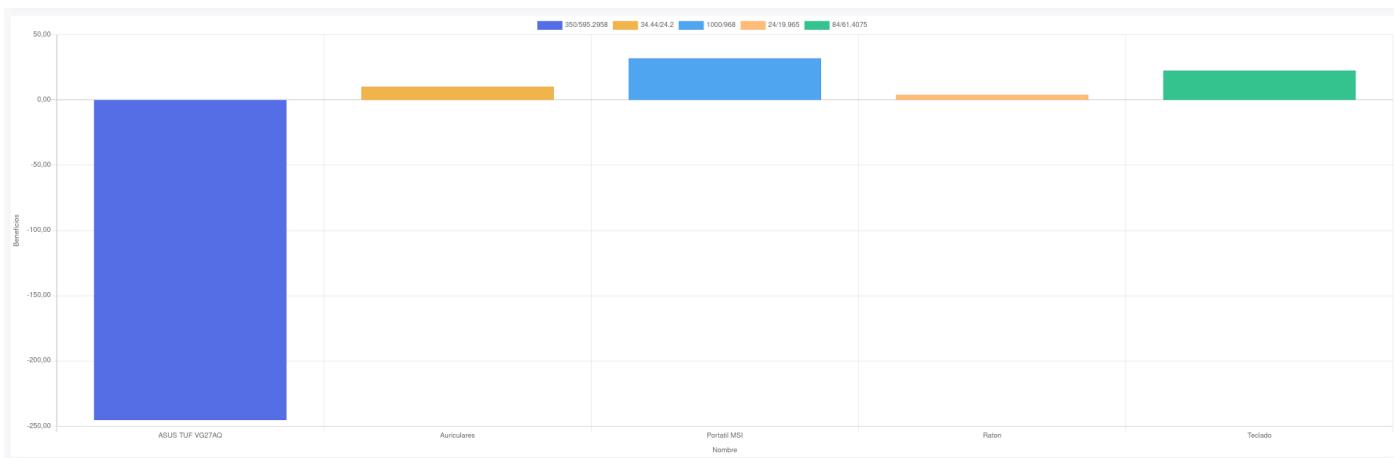
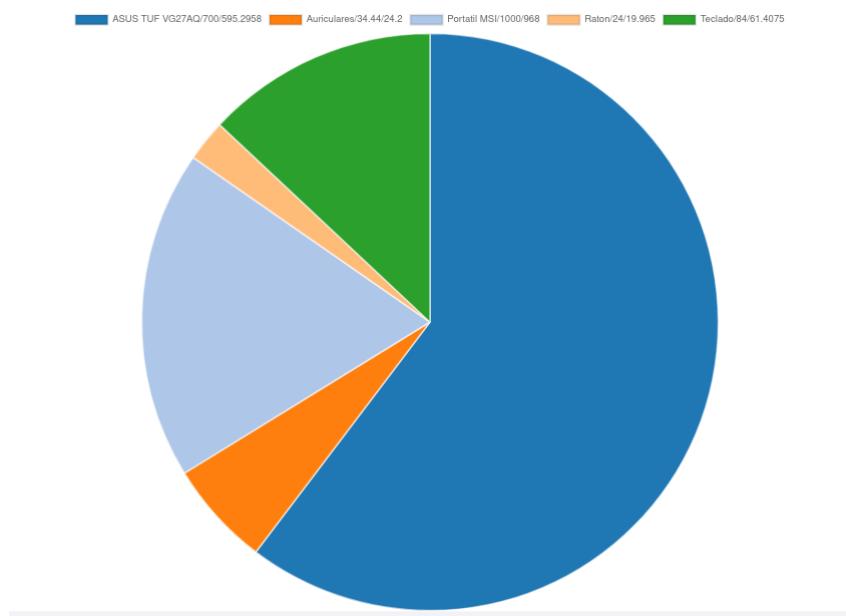


Gráfico Circular: Aviso para poder utilizar este grafico los valores deben ser todos positivos o todos negativos sino no mostrará el grafico circular y dará error. En mi caso como se puede observar en las imágenes anteriores estoy representando los beneficios de una forma gráfica en los esquemas anteriores, pero como tengo un valor negativo como beneficio no me mostrara el diagrama circular por tanto tendré que convertir ese valor negativo en positivo, para ello tendré que realizar la venta del producto **ASUS TUF VG27AQ** que me queda en stock.



Informe del Modelo

Finalmente, en este modelo también he implementado un informe de stock donde se podrá observar todos las características y valores referentes a los productos que tenemos inventariados, para acceder al informe del stock deberemos seleccionar las casillas de cada producto inventariado que deseamos mostrar en el informe, una vez seleccionadas las casillas deberemos clicar sobre el botón imprimir y seleccionar la opción Stock de Productos para poder acceder al informe de stock.

He implementado este informe para gestionar el inventariado de los productos y para el análisis de viabilidad de los productos que distribuimos.



Moracorp Electronics
Plaza España
46960 Aldaya
España

Stock del Producto

Identificador	Numero de Serie	Marca	Nombre del Producto	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Publico(PVP)	Cantidad	Compra	Venta	Ingresos	Costes	Beneficios
81	5	Asus	ASUS TUF VG27AQ	350,0	21,0	423,5	0	2	2	700,0	595,2958	104,7042

6921002890 moracorp.electronics17@gmail.com

Página: 1/1

Modelo de Proveedores

Este modelo se encargará de la gestión y la administración de los proveedores que tiene la empresa. El modelo de proveedores permitirá registrar los proveedores que suministran los productos a la empresa.

☰ Moracorp Electronics Productos Compras Ventas Stock Proveedores Clientes Pedidos Transaccion

Administrator (Moracorp_Electronics)

Proveedores de Moracorp Electronics

ID	Código de Identificación Fiscal(CIF)	Nombre del Proveedor	Foto	Fecha de Fundación	Tipo Sociedad	Dirección del Proveedor	Código Postal	Teléfono	Email	Sede Central
2	1	Asus		19/01/1989	Sociedad Anónima (S.A.)	C/ Metalurgia 38, Planta 1, 08038 Barcelona	08038	932 93 81 54	asus@gmail.com	Taipei,Taiwan
3	2	MSI		17/01/1986	Sociedad Anónima (S.A.)	Avinguda Diagonal, 468, 08006 Barcelona	08006	931 845 542	msi@gmail.com	Taipei,Taiwan

Relaciones del Modelo

Los modelos de productos y compras se relacionan con el modelo de Proveedores para saber quién es el distribuidor del producto en cuestión.

Restricciones del Modelo

Este modelo tiene tres restricciones, la primera es que el CIF del proveedor no se puede repetir y la segunda es que el campo de tipo de sociedad no puede estar vacío se debe seleccionar una de las opciones que te brinda el modelo para poder llevar a cabo la creación del proveedor.

Y la tercera y última restricción es que la fecha de fundación de la empresa del proveedor no puede ser posterior o igual a la fecha actual.

Funcionamiento del Modelo

Una vez tengamos en cuenta las restricciones y relaciones comenzaremos la creación de un proveedor para ello clicamos sobre el botón crear, al clicar el botón se nos abrirá un formulario que deberemos llenar con toda la información relevante del proveedor.

Código de identificación Fiscal(cif)	3	Nombre del Proveedor	Dell
Foto		Fecha de Fundación	01/02/1984
Tipo Sociedad	Sociedad Anónima (S.A.)	Dirección del Proveedor	Ribera del Loira, 8, 28042 Madrid
Código Postal	28042	Teléfono	900 827 780
Email	dell@gmail.com	Sede Central	Round Rock, Texas, Estados Unidos

<input type="checkbox"/> 42	<input type="checkbox"/> 3	Dell		01/02/1984	Sociedad Anónima (S.A.)	Ribera del Loira, 8, 28042 Madrid	28042	900 827 780	dell@gmail.com	Round Rock, Texas, Estados Unidos
-----------------------------	----------------------------	------	---	------------	-------------------------	-----------------------------------	-------	-------------	----------------	-----------------------------------

Si editamos el proveedor que hemos hecho, automáticamente en el modelo de Productos (modificará el campo marca) y Compras (modificará el campo nombre del proveedor) se actualizará el valor referente al distribuidor.

Si borramos la venta también se actualizará automáticamente el valor referente al distribuidor tanto en modelo Productos como en el modelo de Compras.

Vistas del Modelo

Finalmente, en este modelo he aplicado dos vistas la Tree y la Kanban.

La vista Tree nos permitirá ver la información más agrupada y detallada de los proveedores que tenemos asociados con la empresa.

<input type="checkbox"/>	ID	Código de identificación Fiscal(cif)	Nombre del Proveedor	Foto	Fecha de Fundación	Tipo Sociedad	Dirección del Proveedor	Código Postal	Teléfono	Email	Sede Central
<input type="checkbox"/>	2	1	Asus		19/01/1989	Sociedad Anónima (S.A.)	C/ Metalúrgica 38, Planta I, 08038 Barcelona	08038	932 93 81 54	asus@gmail.com	Taipei,Taiwan
<input type="checkbox"/>	3	2	MSI		17/01/1986	Sociedad Anónima (S.A.)	Avinguda Diagonal, 468, 08006 Barcelona	08006	931 845 542	msi@gmail.com	Taipei,Taiwan
<input type="checkbox"/>	42	3	Dell		01/02/1984	Sociedad Anónima (S.A.)	Ribera del Loira, 8, 28042 Madrid	28042	900 827 780	dell@gmail.com	Round Rock, Texas, Estados Unidos

La vista Kanban nos permitirá ver la información de los proveedores de una forma más gráfica e intuitiva.

CIF 1	CIF 2	CIF 3
 <p>Nombre: Asus Direccion: C/ Metalurgia 38, Planta 1, 08038 Barcelona Ciudad: Taipei,Taiwan C.P: 08038 Telefono: 932 93 81 54 Email: asus@gmail.com</p>	 <p>Nombre: MSI Direccion: Avinguda Diagonal, 468, 08006 Barcelona Ciudad: Taipei,Taiwan C.P: 08006 Telefono: 931 845 542 Email: msi@gmail.com</p>	 <p>Nombre: Dell Direccion: Ribera del Loira, 8, 28042 Madrid Ciudad: Round Rock, Texas, Estados Unidos C.P: 28042 Telefono: 900 827 780 Email: dell@gmail.com</p>

Modelo de Clientes

Este modelo se encargará de la gestión y la administración de los clientes a los que distribuyen los productos la empresa. El modelo de clientes permitirá registrar los clientes a los que se les ha vendido y distribuido un producto de nuestra empresa.

Clientes de Moracorp Electronics										
Crear		Listado								
<input type="text" value="Buscar..."/> Filtros Agrupar por ★ Favoritos										
ID	Número de identificación Fiscal(NIF)	Nombre	Foto	Fecha de Registro	Dirección	Teléfono	Email	Código Postal	Localización	
3	1	Alvaro		06/05/2022	Plaza España	692100284	al7mora@gmail.com	46960	Aldaya	Ver
4	3	Miguel		02/05/2022	Calle Atocha	(46) 646 - 564 - 56...	miki@gmail.com	46960	Aldaya	Ver
7	4	Angeles		02/05/2022	Calle Ademuz	692100297	angeles@gmail.com	46001	Valencia	Ver

Relaciones del Modelo

Los modelos de Ventas y Pedidos se relacionan con el modelo de Clientes para saber quién es el consumidor de nuestros productos.

Restricciones del Modelo

Este modelo cuenta con dos restricciones, la primera restricción es que no se puede repetir el NIF del cliente y la segunda restricción es que la fecha de registro no puede ser posterior a la fecha actual.

Funcionamiento del Modelo

Una vez tenemos en cuenta las restricciones y relaciones comenzaremos la creación de un cliente para ello clicamos sobre el botón crear, al clicar el botón se nos abrirá un formulario que deberemos llenar con toda la información relevante del cliente.

Numero de Identificación Fiscal(Nif)	<input type="text" value="5"/>	Nombre	<input type="text" value="Blas"/>
Foto		Fecha de Registro	<input type="text" value="01/05/2022"/>
Dirección	<input type="text" value="Calle del Coscollar"/>	Teléfono	<input type="text" value="692101284"/>
Email	<input type="text" value="blas@gmail.com"/>	Código Postal	<input type="text" value="23001"/>
Localización	<input type="text" value="Jaén"/>		

Below the form, there is a preview section showing the filled-in data:

<input type="checkbox"/>	35	5	Blas	01/05/2022	Calle del Coscollar	692101284	blas@gmail.com	23001	Jaén

Si editamos el cliente que hemos hecho, automáticamente en el modelo de Ventas (modificará el campo cliente) y Pedidos (modificará los diversos campos personales del cliente que hay en pedidos) se actualizará los valores referentes al cliente.

Si borramos el cliente también se actualizará automáticamente los valores referentes al cliente tanto en modelo Ventas como en el modelo de Pedidos.

Vistas del Modelo

Finalmente, en este modelo he aplicado dos vistas la Tree y la Kanban.

La vista Tree nos permitirá ver la información más agrupada y detallada de los clientes a los que les hemos vendido unos productos.

ID	Número de Identificación Fiscal(Nif)	Nombre	Foto	Fecha de Registro	Dirección	Teléfono	Email	Código Postal	Localización
<input type="checkbox"/>	3 1	Alvaro		06/05/2022	Plaza España	692100284	al7mora@gmail.com	46960	Aldaya
<input type="checkbox"/>	4 3	Miguel		02/05/2022	Calle Atocha	(46) 646 - 564 - 56...	miki@gmail.com	46960	Aldaya
<input type="checkbox"/>	7 4	Angeles		02/05/2022	Calle Ademuz	692100297	angeles@gmail.com	46001	Valencia
<input type="checkbox"/>	35 5	Blas		01/05/2022	Calle del Coscollar	692101284	blas@gmail.com	23001	Jaén

La vista Kanban nos permitirá ver la información de los clientes de una forma más gráfica e intuitiva.

NIF 1



Nombre: Alvaro
Dirección: Plaza España
Ciudad: Aldaya
C.P: 46960
Teléfono: 692100284
Email: all7mora@gmail.com

NIF 3



Nombre: Miguel
Dirección: Calle Atocha
Ciudad: Aldaya
C.P: 46960
Teléfono: (45) 646 - 564 - 56
Email: miki@gmail.com

NIF 4



Nombre: Angeles
Dirección: Calle Ademuz
Ciudad: Valencia
C.P: 46001
Teléfono: 692100297
Email: angeles@gmail.com

NIF 5



Nombre: Blas
Dirección: Calle del Coscolar
Ciudad: Jaen
C.P: 23001
Teléfono: 692101284
Email: blas@gmail.com

Modelo de Pedidos

Este modelo se encargará de la gestión y de la administración de los pedidos de los clientes que han realizado en nuestra empresa. El modelo de Pedidos permitirá registrar, tramitar y distribuir los pedidos de nuestra empresa a los clientes haciendo uso de una empresa de mensajería quién llevará los pedidos a la dirección del cliente.

Moracorp Electronics
Productos Compras Ventas Stock Proveedores Clientes Pedidos Transacción
Administrator (Moracorp_Electronics)

Pedidos de Moracorp Electrics

<input type="checkbox"/>	ID	Nº de Pedido	Identificador de la Venta	Distribuidor	NIF del Cliente	Nombre del Cliente	Dirección del Cliente	Teléfono del Cliente	Email del Cliente	Empresa de Logística	Fecha de Expedición	Fecha de Recepción	Estado
<input type="checkbox"/>	3	1	1	Moracorp Electronics	1	Alvaro	Plaza España	692100284	all7mora@gmail.com	SEUR	03/05/2022	03/05/2022	En el Almacén
<input type="checkbox"/>	4	2	2	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	UPS	02/05/2022	12/05/2022	En Reparto
<input type="checkbox"/>	5	3	3	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	SEUR	02/05/2022	11/05/2022	En Reparto
<input type="checkbox"/>	6	4	4	Moracorp Electronics	3	Miguel	Calle Atocha	(45) 646 - 564 - 56	miki@gmail.com	UPS	18/05/2022	19/05/2022	Entregado
<input type="checkbox"/>	7	5	5	Moracorp Electronics	1	Alvaro	Plaza España	692100284	all7mora@gmail.com	SEUR	02/05/2022	11/05/2022	Entregado

Relaciones del Modelo

Este modelo está relacionado con el modelo de Ventas por el campo identificador de la venta, por lo cual debe existir previamente una venta para poder realizar la creación del pedido o se puede crear dicha venta durante la creación del pedido.

Restricciones y Utilidades del Modelo

Este modelo cuenta con tres restricciones, la primera restricción es que el número de pedido no se puede repetir, la segunda restricción es que la fecha de entrega no puede ser anterior a la fecha de expedición.

Y la tercera y última restricción es que el campo estado no puede estar vacío se debe seleccionar una de las opciones que te brinda el modelo para poder llevar a cabo la creación del pedido.

Este modelo cuenta con dos utilidades, la primera utilidad es que tiene una serie de campos calculados (que hacen referencia a los datos del cliente al cual se le ha hecho la venta) que se rellenan automáticamente en función al identificador de la venta, esta utilidad nos puede ser muy beneficiosa porque sólo con indicarle el identificador de la venta te rellena más de la mitad de campos del modelo, y así ahorraremos trabajo al usuario en llenar campos cuya información ya está dentro del ERP y además evitamos posibles errores que pueda tener el usuario al introducir los datos.

38

La segunda y última utilidad es que en función al estado del pedido cambiará el color de la letra de todos los campos del modelo, esta utilidad nos puede ser beneficiosa para tener una guía visual del estado del pedido que tenemos que distribuir, esta guía visual facilita la obtención de la información y por tanto también la eficiencia.

Leyenda de la Guía Visual:

- █ → El color Rojo representa que el pedido está en el Almacén.
- █ → El color Azul representa que el pedido está en Reparto.
- █ → El color Verde representa que el pedido está Entregado.

Funcionamiento del Modelo

Una vez tenidas en cuenta las relaciones y restricciones comenzaremos la creación de un pedido para ello **clicamos sobre el botón crear**, al clicar el botón se nos abrirá un formulario que deberemos llenar con toda la información relevante al pedido.

Moracorp Electronics Productos Compras Ventas Stock Proveedores Clientes Pedidos Transaccion

Pedidos de Moracorp Electrics / Nuevo

Guardar Descartar

Nº de Pedido	6	Identificador de la Venta	7
Distribuidor	Moracorp Electronics	NIF del Cliente	1
Nombre del Cliente	Alvaro	Dirección del Cliente	Plaza España
Teléfono del Cliente	692100284	Email del Cliente	al17mora@gmail.com
Empresa de Logística	DHL	Fecha de Expedición	23/05/2022
Fecha de Recepción	26/05/2022	Estado	Entregado

Como en todos los demás **modelos se puede editar y borrar el pedido de venta que hemos realizado para nuestros clientes**.

Vistas del Modelo

Finalmente, en este modelo he aplicado tres vistas la Tree, la Calendar y la Search.

La vista Tree nos permitirá ver la información más agrupada y detallada de los pedidos de venta que vamos a distribuir a nuestros clientes.

ID	Nº de Pedido	Identificador de la Venta	Distribuidor	NIF del Cliente	Nombre del Cliente	Dirección del Cliente	Teléfono del Cliente	Email del Cliente	Empresa de Logística	Fecha de Expedición	Fecha de Recepción	Estado
3	1	1	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	SEUR	03/05/2022	03/05/2022	En el Almacén
4	2	2	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	UPS	02/05/2022	12/05/2022	En Reparto
5	3	3	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	SEUR	02/05/2022	11/05/2022	En Reparto
6	4	4	Moracorp Electronics	3	Miguel	Calle Atocha	(45) 646 - 564 - 56	miki@gmail.com	UPS	18/05/2022	19/05/2022	Entregado
7	5	5	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	SEUR	02/05/2022	11/05/2022	Entregado
32	6	7	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	DHL	23/05/2022	26/05/2022	Entregado

La vista Calendar nos permitirá observar en un calendario todos los pedidos de venta como eventos en un calendario. Esta vista nos permitirá observar de una manera más sencilla los plazos de entrega de los pedidos y la información más relevante del pedido como la información del cliente, qué empresa de mensajería está distribuyendo el pedido y el estado de este (Si está en el Almacén, En Reparto o Entregado).

Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
17	25	26	27	28	29	30
18	2	3	4	5	6	7
19	9	10	11	12	13	14
20	16	17	18	19	20	21
21	23	24	25	26	27	28
22	30	31			3	4
						5

La vista Search nos brindara la oportunidad de poder filtrar la información por los valores de los campos del modelo, y así poder conseguir la información que deseamos de los pedidos de venta. En este modelo haré el filtrado de la información de los registros de pedidos de venta haciendo uso del campo estado cuyos valores pueden ser En el Almacén, En Reparto o Entregado.

Por tanto, creare un filtro por cada estado que pueda tener el pedido. Para poder aplicar dichos filtros y poder filtrar la información tendremos que hacer clic sobre el botón filtros y seleccionar la opción del estado por la cual queremos filtrar (En el Almacén, En Reparto o Entregado).

Filtrar por el valor En el Almacén del campo estado del pedido:

ID	Nº de Pedido	Identificador de la Venta...	Distribuidor	NIF del Cliente	Nombre del Cliente	Dirección del Cliente	Teléfono del Cliente	Email del Cliente	Empresa de Logística	Fecha de Expedición	Fecha de Recepción	Estado
3	1	1	Moracorp Electronics	1	Alvaro	Plaza España	692100284	all7mora@gmail.com	SEUR	03/05/2022	03/05/2022	En el Almacén

Filtrar por el valor En Reparto del campo estado del pedido:

ID	Nº de Pedido	Identificador de la Venta	Distribuidor	NIF del Cliente	Nombre del Cliente	Dirección del Cliente	Teléfono del Cliente	Email del Cliente	Empresa de Logística	Fecha de Expedición	Fecha de Recepción	Estado
4	2	2	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	UPS	02/05/2022	12/05/2022	En Reparto
5	3	3	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	SEUR	02/05/2022	11/05/2022	En Reparto

Filtrar por el valor Entregado del campo estado del pedido:

Pedidos de Moracorp Electrics													
Pedidos Entregados													
Filtrar													
ID	Nº de Pedido	Identificador de la Venta	Distribuidor	NIF del Cliente	Nombre del Cliente	Direccion del Cliente	Telefono del Cliente	Email del Cliente	Empresa de Logistica	Fecha de Expedicion	Fecha de Recepcion	Estado	
<input type="checkbox"/>	6	4	Moracorp Electronics	3	Miguel	Calle Atocha	(45) 646 - 564 - 56	miki@gmail.com	UPS	18/05/2022	19/05/2022	Entregado	
<input type="checkbox"/>	7	5	Moracorp Electronics	1	Alvaro	Plaza Espana	692100284	al17mora@gmail.com	SEUR	02/05/2022	11/05/2022	Entregado	
<input type="checkbox"/>	32	6	Moracorp Electronics	1	Alvaro	Plaza Espana	692100284	al17mora@gmail.com	DHL	23/05/2022	26/05/2022	Entregado	

Informe del Modelo

Finalmente en este modelo he implementado un informe de albarán de pedidos, donde se podrá observar todos las características y valores referentes a los pedidos que tenemos que distribuir y a los clientes asociados a esos pedidos, para acceder al informe de albarán de pedidos deberemos seleccionar las casillas de cada pedido que deseamos mostrar en el informe, una vez seleccionadas las casillas deberemos clicar sobre el botón imprimir y seleccionar la opción Albarán del Pedido para poder acceder al informe.

He creado este informe para facilitar la distribución y la comunicación entre nuestra empresa y la empresa de mensajería que será quien lleve el producto hasta el cliente.



Moracorp Electronics
Plaza Espana
46960 Aldaya
España

Albaran del Pedido

Datos del Pedido:

Numero de Pedido: 6

Identificador de la Venta: 7

Datos del Cliente:

NIF: 1

Nombre: Alvaro

Direccion: Plaza Espana

Telefono: 692100284

Email: al17mora@gmail.com

Empresa de Logistica	Fecha de Expedicion	Fecha de Recepcion	Estado
DHL	23/05/2022	26/05/2022	Entregado

6921002890 moracorp.electronics17@gmail.com

Página: 1/1

Modelo de Transacción de Venta

Este **modelo se encargará de la gestión y de la administración de las transacciones de venta de los distintos productos que distribuye la empresa**. El **modelo de transacción de venta** nos permitirá registrar en el ERP de la empresa todas las transacciones de venta que se hagan de los distintos productos que suministran a sus clientes.

ID	Producto	Cantidad en stock	Cantidad de Productos Vendidos
1	Teclado	0	3
2	Raton	0	1
3	Teclado	0	1
4	Portatil MSI	0	1
5	Raton	0	1
6	Raton	0	1
7	Portatil MSI	0	1
8	Portatil MSI	0	1
9	Teclado	0	2
10	Raton	0	1
11	Teclado	0	1
12	Portatil MSI	0	2
13	Auriculares	0	1
48	ASUS TUF VG27AQ	0	2

Relaciones del Modelo

El **modelo de Transacción de Venta es un modelo intermedio** porque **hereda los valores del modelo de stock** y es apoyado por el **modelo de ventas para poder crear una venta**, todo esto quiere decir que el **modelo de Transacción de Venta** se relaciona con el **modelo de Stock** mediante la herencia y que el **modelo de Ventas** se relaciona con una **relación muchas a muchos (Many2many)** con el **modelo de transacción de venta**.

Este **modelo nos permite poder vender una gran cantidad de productos desde el modelo de ventas**, si este **modelo no estuviera implementado no se podría realizar la venta de más de un tipo de producto en una misma venta**. Básicamente esto quiere decir que **cada vez que creamos una venta creamos varias transacciones de ventas que vienen reguladas en función a los tipos de productos** (me refiero a productos distintos por ejemplo si en una venta vendemos un ratón y un teclado se hará una transacción de venta para el ratón y otra para el teclado) que **registremos en la venta**.

Restricciones del Modelo

En este **modelo solo hay una restricción** la cual es que la cantidad de producto vendido que deseamos tramitar como **transacción de venta** no puede ser menor o igual a cero.

Funcionamiento del Modelo

Estas **transacciones de ventas se pueden crear desde este modelo o desde el de Ventas**, lo más habitual sería hacerlo desde el de Ventas. Si creamos la transacción desde el **modelo de Transacción de Ventas** deberemos tener en cuenta las **mismas restricciones de stock** que hemos explicado con anterioridad en el **modelo de Ventas**.

Una vez hayamos **creado la transacción** deberíamos **crear una venta y vincularla con la transacción de venta que hemos creado con anterioridad**.

Como ocurre con los demás **modelos en este se podrá modificar y eliminar las Transacciones de Venta**. Y tanto las **modificaciones como las eliminaciones** repercutirán directamente con las ventas de los productos y en el stock de los productos.

Vistas del Modelo

Finalmente, en este **modelo** he implementado la vista **Tree** para ver la información de las transacciones de venta de un forma más agrupada y detallada.

ID	Producto	Cantidad en Stock	Cantidad de Productos Vendidos
1	Teclado	0	3
2	Raton	0	1
3	Teclado	0	1
4	Portatil MSI	0	1
5	Raton	0	1
6	Raton	0	1
7	Portatil MSI	0	1
8	Portatil MSI	0	1
9	Teclado	0	2
10	Raton	0	1
II	Teclado	0	1
12	Portatil MSI	0	1
13	Auriculares	0	2
48	ASUS TUF VG27AQ	0	2

Informes del Módulo

Un informe sirve para crear una **documentación específica sobre una información** que nos pueda ser útil para **desempeñar una labor o actividad**. A lo largo de la **memoria** he explicado **todos los informes que he implementado y porque los he implementado** en nuestro **ERP**, pero no he explicado como he llevado a cabo el **diseño de los informes**, por ello en este apartado vamos a ver como los he **diseñado**.

Para **elaborar el diseño de los informes** he hecho uso de la librería **Bootstrap**. Bootstrap es un **framework front-end multiplataforma, gratuito y de código abierto** para el **diseño de sitios web y aplicaciones**. El marco de trabajo de **Bootstrap se basa en HTML, CSS y JavaScript (JS)** para facilitar el desarrollo de sitios web y aplicaciones.

Esta librería la he usado principalmente para hacer el **decoración y diseño gráfico** de mis informes.

APIS

En este **módulo** he implementado una serie de **APIS** que nos permitirán observar la información que tiene cada **modelo** e incluso podremos hacer creaciones y modificaciones en los mismos. Las **APIS** estarán ubicadas en **Web Controllers** dentro del **Servidor OODO**. Estas **APIS** serán cruciales para poder hacer la **comunicación entre el módulo de Administración Moracorp** que se encuentra en el servidor **OODO** y la **aplicación de escritorio** hecha con el **Framework front-end Quasar**.

He desarrollado una serie de **APIS** que su única función es mostrarnos toda la información que contienen los **modelos que componen el módulo**. Estas **APIS** las utilizare en **Quasar** para mostrar toda la información de los **modelos** en sus respectivas **secciones en la aplicación de escritorio**, para poder hacer esto desde **Quasar** haremos una serie de **peticiones con el método Get del protocolo HTTP** a las **APIS**.

localhost:8069/gestion/moracorp.productos

JSON	Datos sin procesar	Cabeceras
Guardar Copiar Contraer todo Expandir todo Filtrar JSON		
▼ 0:		
id:	1	
n_serie:	"1"	
nombre:	"Teclado"	
marca:	"Asus"	
descripcion:	"Teclado RGB"	
precio:	"12.00"	
porcentaje_iva:	21	
precio_iva:	"14.52"	
▼ 1:		
id:	2	
n_serie:	"2"	
nombre:	"Raton"	
marca:	"Asus"	
descripcion:	"Raton RGB"	
precio:	"12.00"	
porcentaje_iva:	21	
precio_iva:	"14.52"	
▼ 2:		
id:	3	
n_serie:	"3"	
nombre:	"Portatil MSI"	
marca:	"MSI"	
descripcion:	"Portatil I7 con 16 Gb de Ram y disco duro SDD de 512 Gb"	
precio:	"500.00"	
porcentaje_iva:	21	
precio_iva:	"605.00"	

También he implementado una API Rest que nos permitirá poder observar un registro de un modelo en específico. Además, nos permitirá crear un nuevo registro en un modelo en particular. Y incluso nos brindara la oportunidad de editar y eliminar cualquier registro de un modelo.

□ [localhost:8069/gestion/apirest/moracorp.productos?data={"id":"1"}](http://localhost:8069/gestion/apirest/moracorp.productos?data={)

JSON	Datos sin procesar	Cabeceras
Guardar Copiar Contraer todo Expandir todo Filtrar JSON		
▼ 0:		
id:	1	
n_serie:	"1"	
nombre:	"Teclado"	
foto:	"b'9j/4AA0SkZJRpABAQAAAQ_aLUzY2KJQI9FCQlPsEf/z'"	
descripcion:	"<p>Teclado RGB </p>"	
marca:		
0:	2	
1:	"Asus"	
cantidad:	0	
posible_compra:	7	
posible_venta:	7	
compra:	7	
venta:	7	
precio:	12	
porcentaje_iva:	21	
precio_iva:	14.52	
posible_coste:	61.4075	
posible_ingreso:	84	
ingresos:	84	
costes:	61.4075	
beneficios:	22.5925	
video:	"https://www.youtube.com/embed/GVaez4Kn9CM"	
display_name:	"Teclado"	
create_uid:	[]	
create_date:	"2022-05-08 18:46:42.424683"	
write_uid:	[]	
write_date:	"2022-05-27 12:32:30.985974"	
__last_update:	"2022-05-27 12:32:30.985974"	

Esta API Rest la utilizare en Quasar para llevar a cabo la consulta de un registro de un modelo haciendo uso del método Get del protocolo HTTP y también usare esta API Rest para la creación de un nuevo registro en un modelo en particular para ello haré uso del método Post del protocolo HTTP.

Esta Api Rest también permite editar el registro de un modelo con los métodos Put y Patch del protocolo HTTP y borrar un registro de un modelo con el método Delete del protocolo HTTP, pero para la aplicación hecha en Quasar no he utilizado la API Rest para borrar ni modificar ni los métodos Put, Patch y Delete, esto se debe a problemas con las cabeceras HTTP (CORS) he probado muchas posible soluciones como utilizar en el @http.route del Web Controller de la Api Rest los atributos cors='*' y csrf=False para solventar el problema con el CORS pero no dio resultado, y también intente acceder a los archivos de configuración de OODOO y configurarlos para evitar el problema pero tampoco dio resultado.

La única solución viable que me quedaba era instalar un servidor proxy como Nginx para controlar las peticiones y evitar los problemas con el CORS, pero descarte esa solución por no tener tiempo suficiente para poder hacerlo, así que di prioridad a la funcionalidad de la aplicación, aunque la implementación sea un tanto cuestionable por no ser la correcta. Para solventar el problema me cree dos Apis en unos Web Controllers, una de ellas se encargará de modificar los registros y la otro se encargará de borrar los registros, y en vez de utilizar los métodos adecuados Put o Patch para modificar y Delete para borrar haré uso del método Get para modificar y borrar.

Quasar

Aplicación Moracorp

La aplicación de escritorio hecha con el front-end framework Quasar será el otro pilar fundamental del proyecto junto al módulo personalizado. Esta aplicación interactuará y se comunicará haciendo uso de las APIs que hemos patentado en nuestro Servidor OODO. Esta aplicación denominada Aplicación Moracorp nos permitirá gestionar y administrar el módulo personalizado del Servidor OODO desde cualquier dispositivo electrónico.

ID	Nº de Serie	Nombre	Marca	Descripción	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)
1	1	Tecido	Asus	Tecido RGB	12.00	21	14.52
2	2	Raton	Asus	Raton RGB	12.00	21	14.52
3	3	Portatil MSI	MSI	Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb	500.00	21	605.00
4	4	Auriculares	Asus	QUAD DAD™ ES50 9201	17.22	21	20.44
A1	5	ASUS TUF VG37TA	Asus	Monitor 4K	350.00	21	423.50

Esta aplicación se caracteriza por su elegante diseño, intuitiva interfaz gráfica, funcionalidad y por su portabilidad. Esta aplicación será crucial para este proyecto porque nos permitirá llevar a cabo gran parte de las acciones y funcionalidades del módulo de Administración Moracorp desde la aplicación de escritorio, todo esto significa que podremos gestionar desde la aplicación los productos y el stock de los mismos, las compras y ventas de productos, la tramitación y administración de los pedidos y también se encargará de dar un buen soporte en las comunicaciones y relaciones con los clientes como con los distribuidores o proveedores.

Esta aplicación de escritorio será multiplataforma porque sera compatible con los sistemas operativos macOS, Windows y Linux.

Librerías implementadas en la Aplicación

Apexcharts es una librería de gráficos de código abierto con licencia del MIT que ayuda a los desarrolladores a crear visualizaciones atractivas e interactivas de gráficos para páginas web y aplicaciones. Esta librería la he usado para crear un gráfico de barras donde se podía observar de una forma muy detallada y gráfica los costes, ingresos y beneficios que nos proporcionan cada producto de la empresa.

Axios es una librería de Javascript que se utiliza para realizar solicitudes HTTP desde Node.js o XMLHttpRequests desde el navegador y es compatible con la API de Promise que es nativa de JS ES6. Se puede usar para interceptar solicitudes y respuestas HTTP y permite la protección del lado del cliente contra XSRF.

También tiene la capacidad de cancelar solicitudes. La librería Axios nos proporciona las siguientes ventajas que serán fundamentales para la elaboración de este proyecto:

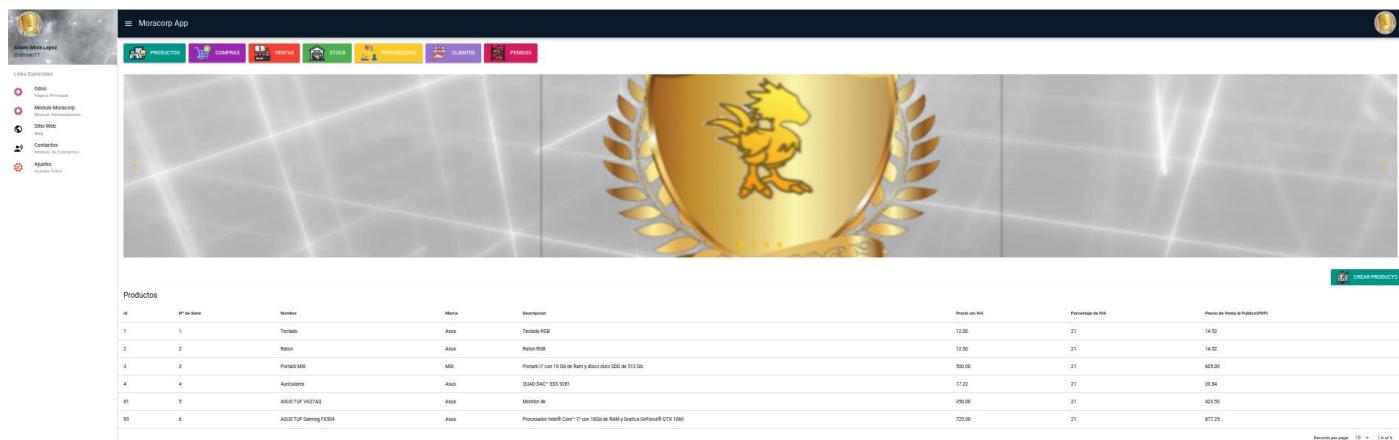
- Axios ofrece una API unificada para las solicitudes Ajax.
- Esta librería está pensada para facilitar el consumo de servicios web, API REST que devuelvan datos JSON.

- Es muy sencilla de usar y puede ser un complemento ideal para sitios web convencionales, donde no se esté usando **jQuery** y aplicaciones front-end modernas basadas en **librerías como React o Polymer**, que no disponen en su **Core** de mecanismos para hacer de cliente **HTTP**.
- Ocupa poco espacio en el disco duro apenas pesa **13 Kb**.
- Esta librería nos ahorra una cantidad notable de trabajo de codificación para realizar solicitudes **HTTP**.
- Axios tiene una gran compatibilidad con la mayoría de los navegadores que están en el mercado.

La librería Axios será fundamental para nuestra aplicación de escritorio porque será la encargada de realizar las solicitudes a las APIS que tenemos implementadas, esta librería nos permitirá observar la información de los modelos del módulo de Administración Moracorp y además nos brindará la oportunidad de crear, editar y borrar registros de nuestros modelos.

Menú de la Aplicación

El menú de la aplicación nos dará una serie de opciones que nos facilitaran el acceso a los recursos principales del Servidor OODOO desde la aplicación de escritorio. Para acceder al menú de la aplicación tendremos que hacer clic sobre el icono que tiene tres rayas que se encuentra en la cabecera de la aplicación justo al lado del título del software.



El menú de la aplicación cuenta con las siguientes opciones:



- La primera opción del menú nos redireccionara a la página principal del Servidor OODOO.
- La segunda opción del menú nos redireccionara al módulo personalizado de Administración de Moracorp que he implementado en ERP.
- La tercera opción del menú nos redireccionara al sitio web que he creado en el Servidor OODOO.
- La cuarta opción del menú nos redireccionara al módulo contactos de nuestro ERP.
- Y la quinta y última opción nos redireccionara a la página de configuración o ajustes del Servidor OODOO.

Sección de Productos

En esta sección de la aplicación podremos observar todos los valores de los productos registrados y también podremos crear, editar y eliminar registros referentes a los productos. Toda acción que llevemos a cabo repercutirá en nuestro módulo personalizado del servidor ODOO.

ID	Nº de Serie	Nombre	Marca	Descripción	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)
1	1	Teclado	Asus	Teclado RGB	12.00	21	14.52
2	2	Raton	Asus	Raton RGB	12.00	21	14.52
3	3	Portatil MSI	MSI	Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb	500.00	21	605.00
4	4	Auriculares	Asus	QUAD DAC™ ESS 9281	17.22	21	20.84
81	5	ASUS TUF VG27AQ	Asus	Monitor 4k	350.00	21	423.50

Si clicamos sobre el botón crear producto se nos abrirá una página con un formulario que deberemos llenar con los valores que deseamos que tenga el nuevo producto teniendo en cuenta las restricciones y relaciones que tiene el modelo de Productos del módulo de Administración de Moracorp. Como tenemos las restricciones del modelo Productos he tenido que volverlas a implementar en la aplicación para que el usuario que está usando la aplicación se dé cuenta de cuáles son los errores que ha cometido y los solventa antes de crear el nuevo registro, en el caso de que no solvete su error no se permitirá la creación del nuevo registro hasta que se solventen los fallos.

Introduce valores a los Campos para Crear un Producto:

ID	Nº de Serie	Nombre	Marca	Descripción	Precio sin IVA	Porcentaje de IVA	Precio de Venta al Público(PVP)
1	1	Teclado	Asus	Teclado RGB	12.00	21	14.52
2	2	Raton	Asus	Raton RGB	12.00	21	14.52
3	3	Portatil MSI	MSI	Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb	500.00	21	605.00
4	4	Auriculares	Asus	QUAD DAC™ ESS 9281	17.22	21	20.84
81	5	ASUS TUF VG27AQ	Asus	Monitor 4k	350.00	21	423.50
85	6	ASUS TUF Gaming FX504	Asus	Procesador Intel® Core™ i7 con 16Gb de RAM y Grafica GeForce® GTX 1060	725.00	21	877.25

Nada más crearse el nuevo producto se actualizará el contenido del modelo Productos y a su vez también se actualizará el modelo stock añadiendo dicho producto al stock.

85	6	ASUS TUF Gaming FX504	Asus	Procesador Intel® Core™ i7 con 16Gb de RAM y Grafica GeForce® GTX 1060	725,00	21,00	877,25
----	---	-----------------------	------	--	--------	-------	--------

Si nos dirigimos a la **tabla de productos** y hacemos clic sobre la fila de un producto se nos abrirá una página donde podremos ver las **características fundamentales del producto y una imagen y un video** de este.

The screenshot shows the Moracorp App interface. At the top, there's a navigation bar with icons for PRODUCTOS, COMPRAS, VENTAS, STOCK, PROVEEDORES, CLIENTES, and PEDIDOS. Below this, a sub-menu bar shows 'Moracorp App' on the left and a gold coin icon on the right. The main content area displays product details for an 'ASUS TUF Gaming FX504'. It includes the product name, brand ('Asus'), description ('Procesador Intel® Core™ i7 con 16Gb de RAM y Grafica GeForce® GTX 1060'), and price ('PVP: 877,25'). To the left of the text is a thumbnail image of the laptop. Below the text is a large, vibrant image of the laptop set against a dark, star-filled space background with a bright orange planet and glowing red energy beams.

Si quisiéramos **editar un producto** nos dirigimos a la **tabla de productos** y clicamos sobre la fila del producto, una vez hayamos clicado se nos **abrirá una página** donde se podrá ver la **información principal del producto**, cuando estemos **dentro de la página** deberemos hacer **clic sobre el botón que tiene el icono de un lápiz y es de color amarillo** para poder **editar el producto**. Justo después de haber clicado en el botón se nos **abrirá un formulario donde podremos editar el producto**, pero teniendo en cuenta **las restricciones y relaciones del modelo de Productos** porque si no se atiende a las restricciones no se podrá editar el producto.

The screenshot shows the 'Editar Producto' (Edit Product) form. At the top, there's a navigation bar with icons for PRODUCTOS, COMPRAS, VENTAS, STOCK, PROVEEDORES, CLIENTES, and PEDIDOS. Below this, a sub-menu bar shows 'Moracorp App' on the left and a gold coin icon on the right. The main content area is titled 'Editar Producto:' and contains a form with the following fields:
- Nº de Serie: 6
- Nombre: ASUS TUF Gaming FX504
- Marca: 2
- Descripción: Procesador Intel® Core™ i7 con 16Gb de RAM y Grafica GeForce® GTX 1060
- Precio sin IVA: 725
- Porcentaje de IVA: 21
- URL del Video: <https://www.youtube.com/embed/7Ukstsu-dY>
At the bottom of the form is a blue 'EDITAR' (Edit) button.

Si quisiéramos **eliminar** tendríamos que **seguir los mismos pasos que cuando hemos editado**, pero con la **excepción de que** deberemos **clicar sobre el botón que tiene el icono de la papelera y es de color rojo**. Una vez hayamos clicado el botón el **producto se borrará**.



Nº de Serie: 6

Nombre: ASUS TUF Gaming FX504

Marca: Asus

Descripción: Procesador Intel® Core™ i7 con 16Gb de RAM y Grafica GeForce® GTX 1060

PVP: 877.25



Sección de Compras

En esta **sección de la aplicación** podremos **observar todos los valores de las compras registradas de cada producto** y también podremos **crear, editar y eliminar registros** referentes a las **compras**.

The screenshot shows the Moracorp App interface with the 'Compras' tab selected. The main area displays a table of purchase records:

ID	Nº de Transacción	Nombre del Proveedor	Fecha de Compra	Fecha de Entrega	Producto	Precio por Unidad sin IVA	Porcentaje de IVA	Precio por Unidad con IVA	Cantidad	Precio Total de la Compra
3	1	Asus	2022-05-08	2022-05-10	Teclado	7.25	21	8.77	6	52.63
4	2	Asus	2022-05-10	2022-05-14	Raton	8.25	21	9.98	2	19.96
5	3	MSI	2022-05-09	2022-05-21	Portatil MSI	400.00	21	484.00	2	968.00
6	4	Asus	2022-05-02	2022-05-27	Teclado	7.25	21	8.77	1	8.77
7	5	Asus	2022-05-09	2022-05-12	Auriculares	10.00	21	12.10	2	24.20
126	6	Asus	2022-05-09	2022-05-13	ASUS TUF VG27AQ	245.99	21	297.65	2	595.30

At the bottom right, there is a note: "Records per page: 10 ▾ 1-6 of 6".

Si clicamos sobre el botón **crear compra** se nos abrirá una página con un formulario que deberemos **rellenar con los valores que deseamos que tenga la nueva compra de un producto**, teniendo en **cuenta las restricciones y relaciones que tiene el modelo de compras del módulo de Administración de Moracorp**. Deberemos cumplir las restricciones porque en el caso de que no lo hagamos se **impedirá la creación del nuevo registro**.

The screenshot shows the Moracorp App interface with the 'Crear Compra' (Create Purchase) form. The fields are as follows:

- Nº de Transacción: 7
- Nombre del Proveedor: 2
- Fecha de Compra: 25 / 05 / 2022
- Fecha de Entrega: 27 / 05 / 2022
- Producto: 85
- Precio por Unidad sin IVA: 585
- Porcentaje de IVA: 21
- Cantidad: 3

At the bottom center is a 'CREAR' (Create) button.

Compras

ID	Nº de Transaccion	Nombre del Proveedor	Fecha de Compra	Fecha de Entrega	Producto	Precio por Unidad sin IVA	Porcentaje de IVA	Precio por Unidad con IVA	Cantidad	Precio Total de la Compra
3	1	Asus	2022-05-08	2022-05-10	Teclado	7.25	21	8.77	6	52.63
4	2	Asus	2022-05-10	2022-05-14	Raton	8.25	21	9.98	2	19.96
5	3	MSI	2022-05-09	2022-05-21	Portatil MSI	400.00	21	484.00	2	968.00
6	4	Asus	2022-05-02	2022-05-27	Teclado	7.25	21	8.77	1	8.77
7	5	Asus	2022-05-09	2022-05-12	Auriculares	10.00	21	12.10	2	24.20
126	6	Asus	2022-05-09	2022-05-13	ASUS TUF VG27AQ	245.99	21	297.65	2	595.30
127	7	Asus	2022-05-25	2022-05-27	ASUS TUF Gaming FX504	585.00	21	707.85	3	2123.55

Records per page: 10 ▾ 17 of 7

Justo después de **crear la compra de un producto se actualizará el modelo de Compras y el modelo Stock**, en el **modelo Stock** se actualizarán los valores de los **campos cantidad en stock, la cantidad comprada, los costes y los beneficios del producto que ha sido comprado**.

<input type="checkbox"/>	127	7	ASUS	25/05/2022	27/05/2022	ASUS TUF Gaming FX504	585,00	21,00	707,85	3	2123,55
--------------------------	-----	---	------	------------	------------	-----------------------	--------	-------	--------	---	---------

Si nos dirigimos a la **tabla de compras** y hacemos clic sobre la fila de una transacción de compra de un producto se nos **abrirá una página**, donde podremos ver de una forma más organizada y detallada toda la información de la **transacción de compra**.

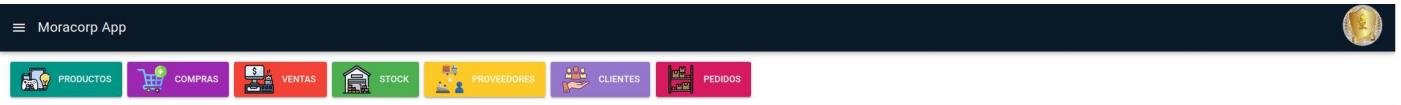
The screenshot shows the Moracorp App interface. At the top is a dark header with the text "Moracorp App" and a gold coin icon. Below the header is a horizontal menu bar with six items: PRODUCTOS (blue), COMPRAS (purple), VENTAS (green), STOCK (yellow), PROVEEDORES (orange), CLIENTES (pink), and PEDIDOS (red). The "COMPRAS" button is highlighted.

Nº de Transaccion: 7

The screenshot shows a detailed view of a purchase transaction. At the top, there are input fields for "Nombre del Proveedor" (Asus), "Fecha de Compra" (25 / 05 / 2022), "Fecha de Entrega" (27 / 05 / 2022), "Producto" (ASUS TUF Gaming FX504), "Precio por Unidad sin IVA" (585.00), "Porcentaje de IVA" (21), "Precio por Unidad con IVA" (707.85), and "Cantidad" (3). Below these fields is a note: "Precio Total de la Compra 2123.55". At the bottom are two buttons: a yellow edit button with a pencil icon and a red delete button with a trash icon.

Una vez dentro de la **página que explica al detalle la compra de un producto** en particular podremos **hacer uso de dos funcionalidades** las cuales **son editar y borrar**, para ello tendremos que **clikar en uno de los botones** para desarrollar **una función u otra**.

Si **clicamos el botón de color amarillo** podremos **editar la compra que hemos realizado de un producto**, pero teniendo siempre en **consideración durante la edición las restricciones y relaciones del modelo Compras**.



Moracorp App

- [!\[\]\(3e6467a616b95d882be4fa9b55ba5e70_img.jpg\) PRODUCTOS](#)
- [!\[\]\(1e9851fd217eb0511ec1424a41c34d85_img.jpg\) COMPRAS](#)
- [!\[\]\(fc0f2685b55edc30551e227637950178_img.jpg\) VENTAS](#)
- [!\[\]\(0bbb4856f764e1911753ec5bba775182_img.jpg\) STOCK](#)
- [!\[\]\(e0e08e599118cc43a691288ea344b6e3_img.jpg\) PROVEEDORES](#)
- [!\[\]\(e54ac33e412a9384cbeedc4735946d48_img.jpg\) CLIENTES](#)
- [!\[\]\(23cf182ca2b9851c13d182b9ccea832a_img.jpg\) PEDIDOS](#)

Editar Compra:

Nº de Transaccion:	7
Nombre del Proveedor:	2
Fecha de Compra:	25 / 05 / 2022
Fecha de Entrega:	27 / 05 / 2022
Producto:	85
Precio por Unidad sin IVA:	585
Porcentaje de IVA:	21
Cantidad:	3

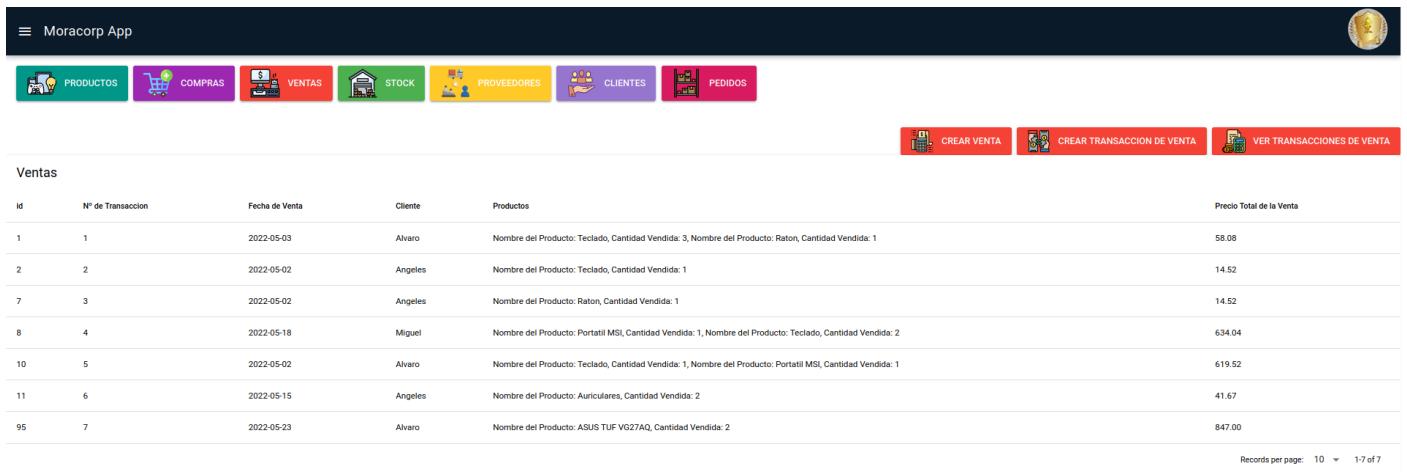
 EDITAR

Si clicamos el botón de color rojo borraremos la transacción de compra que hemos hecho.

Tanto si editamos como eliminamos se actualizará automáticamente el modelo de Compras y el modelo de Stock.

Sección de Ventas y Transacciones de Venta

En esta sección de la aplicación podremos observar todos los valores de las ventas y de las transacciones de venta registradas de los productos, y también podremos crear, editar y eliminar registros referentes a las ventas y transacciones de venta.



Moracorp App

- [!\[\]\(b92a39db1fb35ddc1f6697ca6f2b9b27_img.jpg\) PRODUCTOS](#)
- [!\[\]\(73e4527f22f359ed3e5dc501d53a0d5c_img.jpg\) COMPRAS](#)
- [!\[\]\(beb5a1d4faf2d734b32ad9898056830c_img.jpg\) VENTAS](#)
- [!\[\]\(aee0179e15870d5979d3892b9284c9cc_img.jpg\) STOCK](#)
- [!\[\]\(7e9faeb318dfa228e115af760828dcab_img.jpg\) PROVEEDORES](#)
- [!\[\]\(57f073cc827528b2e4ca301cea6db913_img.jpg\) CLIENTES](#)
- [!\[\]\(e5ac6ce9c79a3ec00e3517d39bbbfed8_img.jpg\) PEDIDOS](#)

[!\[\]\(1893a1d548f358ca04fd69005eda26d5_img.jpg\) CREAR VENTA](#)
[!\[\]\(da0917627a28d12fb8a4658a90ff87f9_img.jpg\) CREAR TRANSACCION DE VENTA](#)
[!\[\]\(55913b821acf839236e0ace5a06e6345_img.jpg\) VER TRANSACCIONES DE VENTA](#)

Ventas					
ID	Nº de Transaccion	Fecha de Venta	Cliente	Productos	Precio Total de la Venta
1	1	2022-05-03	Alvaro	Nombre del Producto: Teclado, Cantidad Vendida: 3, Nombre del Producto: Ratón, Cantidad Vendida: 1	58.08
2	2	2022-05-02	Angeles	Nombre del Producto: Teclado, Cantidad Vendida: 1	14.52
7	3	2022-05-02	Angeles	Nombre del Producto: Ratón, Cantidad Vendida: 1	14.52
8	4	2022-05-18	Miguel	Nombre del Producto: Portátil MSI, Cantidad Vendida: 1, Nombre del Producto: Teclado, Cantidad Vendida: 2	634.04
10	5	2022-05-02	Alvaro	Nombre del Producto: Teclado, Cantidad Vendida: 1, Nombre del Producto: Portátil MSI, Cantidad Vendida: 1	619.52
11	6	2022-05-15	Angeles	Nombre del Producto: Auriculares, Cantidad Vendida: 2	41.67
95	7	2022-05-23	Alvaro	Nombre del Producto: ASUS TUF VG27AQ, Cantidad Vendida: 2	847.00

Records per page: 10 ▾ 1-7 of 7

He unido en esta sección los dos modelos que se encuentran separados en el servidor ODOO, esta decisión de diseño se debe a que ambos modelos dependen el uno del otro para poder realizar las ventas. Todo esto quiere decir que si no creamos una transacción de venta para vincular a la venta no se podrá llevar a cabo la venta correctamente porque sería una venta inútil e ineficiente porque no se vendería ningún producto y lo mismo ocurriría al revés. Por ese motivo he elegido unir estos modelos en una misma sección y además también los he unido porque nos facilita las interacciones de creación, edición y eliminación de las ventas, y gracias a esa facilidad que nos brinda el diseño seremos más eficientes para llevar a cabo nuestra labor.

Comenzaremos la explicación de la sección con las transacciones de Venta.

Transacciones de Venta

Si clicamos sobre el botón **ver transacciones de venta** se nos abrirá una página donde podremos **observar todas las transacciones de Venta** que tenemos registradas en el **modelo de Transacciones de Venta del Servidor OODOO**.

Id	Producto Vendido	Cantidad Vendida		
			Cantidad Vendida	
1	Teclado	3		
2	Raton	1		
3	Teclado	1		
4	Portatil MSI	1		
5	Raton	1		
6	Raton	1		
7	Portatil MSI	1		
8	Portatil MSI	1		
9	Teclado	2		
10	Raton	1		

Records per page: 10 1-10 of 14 < >

Si nos dirigimos a la **tabla de transacciones de venta** y hacemos clic sobre la fila de una transacción de venta de un **producto** se nos abrirá **una página**, donde podremos ver de una forma más organizada y detallada toda la **información de la transacción de venta**.

Identificador de Transaccion:	48
Producto Vendido	ASUS TUF VG27AQ
Cantidad Vendida	2

EDITAR

Una vez dentro de la **página que explica al detalle la transacción de venta de un producto** en particular podremos hacer uso de **dos funcionalidades**, las cuales son **editar y borrar** para ello tendremos que **clikar en uno de los botones para desarrollar una función u otra**.

Si clicamos el **botón de color amarillo** podremos **editar la transacción de venta** que hemos **realizado de un producto**, pero teniendo siempre en **consideración** durante la edición las **restricciones y relaciones** del **modelo de Transacción de Ventas**.

Editar Venta:

Producto Vendido:
81

Cantidad Vendida:
2

EDITAR

Si clicamos el **botón de color rojo** **borraremos la transacción de venta que hemos hecho**.

Tanto si **editamos** como **eliminamos** se **actualizará automáticamente el modelo de Transacción de Ventas, el modelo de Ventas y el modelo de Stock**. Se **actualizará el modelo de Ventas y el modelo de Stock siempre y cuando la transacción haya sido relacionada con una venta porque sino no afectará a esos modelos mencionados.**

Ahora **volveremos** a la **página principal** de esta sección para ello haremos clic en el botón **ventas**.

Una vez de vuelta a la **página principal de la sección** clicaremos sobre el **botón crear transacción de venta**, al clicarlo se nos **abrirá** una **página con un formulario** que deberemos **rellenar** con **los valores** que deseamos que **tenga la nueva transacción de venta de un producto** teniendo en cuenta **las restricciones y relaciones** que tiene el **modelo de Transacción de Venta** del **módulo de Administración de Moracorp**. Deberemos **cumplir las restricciones** porque en el caso de que **no lo hagamos** se **impedirá** la **creación del nuevo registro**.

Introduce valores a los Campos para Crear una Transaccion de Venta:

Producto 85	Cantidad Vendida 3
CREAR	

Justo después de **crear la transacción de venta** se **actualizará el modelo de Transacciones de Venta**.

52 ASUS TUF Gaming FX504

Una vez **creada la transacción de venta** deberemos **vincularla a una venta** para poder llevar a cabo **correctamente la creación de la venta**.

Venta

Si clicamos sobre el **botón crear venta** se nos **abrirá** una **página con un formulario** que deberemos **rellenar** con **los valores** que deseamos que **tenga la nueva venta de un producto**, teniendo en cuenta **las restricciones y relaciones** que tiene el **modelo de Ventas** del **módulo de Administración de Moracorp**. Justo en este momento tendremos que **vincular la transacción de venta** que hicimos anteriormente con la **venta que estamos creando ahora mismo con este formulario**, esto quiere decir que **tendremos que insertar en el campo productos del formulario la relación**, la cual será el **id de la transacción de venta** que hicimos anteriormente. Deberemos **cumplir las restricciones** porque en el caso de que **no lo hagamos** se **impedirá** la **creación del nuevo registro**.

Introduce valores a los Campos para Crear una Venta:

Nº de Transaccion 8	Fecha de Venta 30 / 05 / 2022	Cliente 35	Productos 52
CREAR			

Ventas					
ID	Nº de Transaccion	Fecha de Venta	Cliente	Productos	Precio Total de la Venta
1	1	2022-05-03	Alvaro	Nombre del Producto: Teclado, Cantidad Vendida: 3, Nombre del Producto: Ratón, Cantidad Vendida: 1	58.08
2	2	2022-05-02	Angeles	Nombre del Producto: Teclado, Cantidad Vendida: 1	14.52
7	3	2022-05-02	Angeles	Nombre del Producto: Ratón, Cantidad Vendida: 1	14.52
8	4	2022-05-18	Miguel	Nombre del Producto: Portátil MSI, Cantidad Vendida: 1, Nombre del Producto: Teclado, Cantidad Vendida: 2	634.04
10	5	2022-05-02	Alvaro	Nombre del Producto: Teclado, Cantidad Vendida: 1, Nombre del Producto: Portátil MSI, Cantidad Vendida: 1	619.52
11	6	2022-05-15	Angeles	Nombre del Producto: Auriculares, Cantidad Vendida: 2	41.67
95	7	2022-05-23	Alvaro	Nombre del Producto: ASUS TUF VG27AQ, Cantidad Vendida: 2	847.00
112	8	2022-05-30	Bias	Nombre del Producto: ASUS TUF Gaming FX504, Cantidad Vendida: 3	2631.75

Records per page: 10 ▾ 1-8 of 8

Justo después de **crear la venta de un producto** se **actualizará automáticamente el modelo Ventas y el modelo Stock**, en el **modelo Stock** se **actualizarán los valores de los campos cantidad en stock, venta, ingresos y beneficios de los productos que han sido vendidos en la venta**.

<input type="checkbox"/>	112	8	30/05/2022	35	1 registro	2.631,75
--------------------------	-----	---	------------	----	------------	----------

Si nos dirigimos a la tabla de ventas y hacemos clic sobre la fila de una venta de un producto se nos abrirá una página donde podremos ver de una forma más organizada y detallada toda la información de la venta.

≡ Moracorp App 

PRODUCTOS
COMPRAS
VENTAS
STOCK
PROVEEDORES
CLIENTES
PEDIDOS

Nº de Transaccion: 8

Fecha de Venta 30 / 05 / 2022	Cliente Bias	Precio Total de la Venta 2631.75
Productos Nombre del Producto: ASUS TUF Gaming FX504, Cantidad Vendida: 3		

✎ ✖

Una vez dentro de la **página que explica al detalle la venta de un producto** en particular podremos hacer **uso de dos funcionalidades** las cuales son **editar y borrar**, para ello tendremos que **clikar en uno de los botones para desarrollar una función u otra**.

Si **clicamos el botón de color amarillo** podremos **editar la venta** que hemos realizado de un **producto**, pero teniendo siempre en consideración durante la edición las restricciones y relaciones del **modelo de Ventas**.

≡ Moracorp App 

PRODUCTOS
COMPRAS
VENTAS
STOCK
PROVEEDORES
CLIENTES
PEDIDOS

Editar Venta:

Nº de Transaccion:
 8

Fecha de Venta:
 30 / 05 / 2022

Cliente:
 35

Productos:
 52

✎ ✖

Si clicamos el botón de color rojo borraremos la venta que hemos hecho.

Tanto si editamos y eliminamos una venta de productos se actualizará el modelo Ventas y el modelo Stock.

Sección de Stock

En esta sección de la aplicación podremos analizar el Stock y la viabilidad de los productos del modelo Stock. La sección Stock estará más enfocada a la consulta de la información del stock y de la viabilidad de los productos que en funcionalidades, pero igualmente tendremos funcionalidades como editar y borrar los productos que tenemos inventariados. Y esta sección contará con un gráfico de barras que mostrará los beneficios que nos proporciona cada producto. El modelo Stock no es un modelo como tal porque en realidad forma parte del modelo productos esto significa que ambos son el mismo modelo por ello stock puede llevar a cabo las mismas acciones que el modelo de productos, y por tanto en la aplicación de escritorio en la sección de Stock se podrán llevar a cabo funcionalidades del modelo Productos como editar y borrar productos.

The screenshot shows the 'Stock' section of the Moracorp App. At the top, there's a navigation bar with icons for PRODUCTOS, COMPRAS, VENTAS, STOCK, PROVEEDORES, CLIENTES, and PEDIDOS. A gold trophy icon is on the far right. Below the navigation is a green button labeled 'GRAFICO DE CONTABILIDAD'. The main area displays a table titled 'Stock de Productos' with columns for Id, Nombre, Cantidad en Stock, Compra, Venta, Ingresos, Costes, and Beneficios. The table lists several products like Teclado, Raton, Portatil MSI, Auriculares, ASUS TUF VG27AQ, and ASUS TUF Gaming FX504. At the bottom right of the table, it says 'Records per page: 10 ▾ 1-6 of 6'.

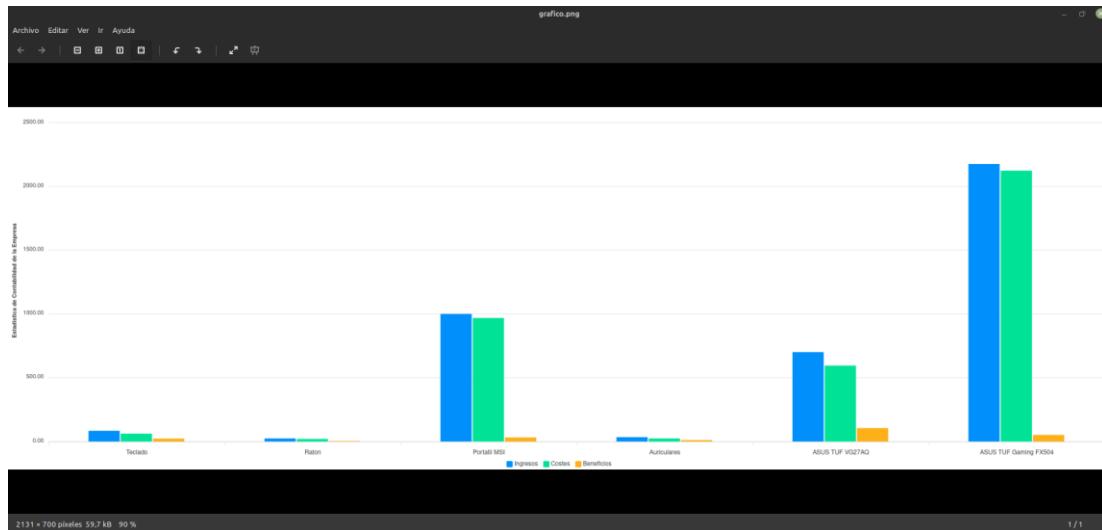
Si clicamos sobre el botón gráfico de contabilidad se nos creará un gráfico de barras a partir de los costes, ingresos y beneficios que nos proporciona cada producto.



Dentro de la página donde está ubicado el gráfico de barras hay una funcionalidad que nos permite descargar la imagen como PNG o JPG, y además también nos brindara la oportunidad de descargar un archivo CSV con la información que está representada gráficamente en el gráfico de barras, esto quiere decir que obtendremos en un archivo CSV toda la información referente a los costes, ingresos y beneficios que nos aportan los productos. Para acceder a esa funcionalidad deberemos clicar sobre el icono que tiene tres rayas negras que está ubicado en la esquina superior derecha del gráfico de barras, y posteriormente seleccionar la opción que deseamos llevar a cabo.



Imagen Descargada:



Archivo CSV Descargado:

grafico.csv - LibreOffice Calc														
Archivo Editar Ver Insertar Formato Estilos Hoja Datos Herramientas Ventana Ayuda														
A1														
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1 category	Ingresos	Costes	Beneficios											
2 Teclado	84 61.41	22.59												
3 Ratón	24 19.96	4.04												
4 Portátil MSI	1000	968	32											
5 Auriculares	34.44	24.2	10.24											
6 ASUS TUF VG27AQ	700 595.3	104.7												
7 ASUS TUF Gaming FX504	2175 2123.55	51.45												
8														

Si nos dirigimos a la tabla de productos inventariados y hacemos clic sobre la fila de un producto se nos abrirá una página donde podremos ver las características fundamentales del producto y una imagen y un video de este.

Moracorp App

Nº de Serie: 3

Nombre: Portatil MSI
Marca: MSI
Descripción: Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb
PVP: 605.00



Una vez dentro de la **página que explica las características fundamentales de un producto** en particular podremos hacer uso de dos funcionalidades las cuales son **editar** y **borrar** para ello tendremos que **clicar en uno de los botones para desarrollar una función u otra.**

Si clicamos el **botón de color amarillo** podremos **editar el producto inventariado que hayamos hecho** con anterioridad, pero teniendo siempre en consideración durante la edición las restricciones y relaciones del **modelo de Stock**.

Moracorp App

Editar Producto:

Nº de Serie:
3

Nombre:
Portatil MSI

Marca:
3

Descripción:
Portatil i7 con 16 Gb de Ram y disco duro SSD de 512 Gb

Precio sin IVA:
500

Porcentaje de IVA:
21

URL del Video:
<https://www.youtube.com/embed/nTW6J0eSpYc>

EDITAR

Si clicamos el **botón de color rojo** **borraremos el producto inventariado** que hayamos hecho.

Tanto si **editamos** como **eliminamos** un producto inventariado se **actualizará el modelo Stock y el modelo Producto**.

Sección de Proveedores

En esta sección de la aplicación podremos observar todos los valores de los proveedores que tenemos registrados y también podremos crear, editar y eliminar registros referentes a los proveedores.

ID	Código de Identificación Fiscal(CIF)	Nombre del Proveedor	Fecha de Fundación	Tipo Sociedad	Dirección del Proveedor	Código Postal	Teléfono	Email	Sede Central
2	1	Asus	1989-01-19	Sociedad Anónima (S.A.)	C/ Metalurgia 38, Planta 1, 08038 Barcelona	08038	932 93 81 54	asus@gmail.com	Taipei,Taiwan
3	2	MSI	1986-01-17	Sociedad Anónima (S.A.)	Avinguda Diagonal, 468, 08006 Barcelona	08006	931 845 542	msi@gmail.com	Taipei,Taiwan
42	3	Dell	1984-02-01	Sociedad Anónima (S.A.)	Ribera del Loira, 8, 28042 Madrid	28042	900 827 780	dell@gmail.com	Round Rock, Texas, Estados Unidos

Records per page: 10 ▾ 1-3 of 3

Si clicamos sobre el botón crear proveedor se nos abrirá una página con un formulario que deberemos rellenar con los valores que deseamos que tenga el nuevo proveedor teniendo en cuenta las restricciones y relaciones que tiene el modelo de Proveedores del módulo de Administración de Moracorp. Deberemos cumplir las restricciones porque en el caso de que no lo hagamos se impedirá la creación del nuevo registro.

Introduce valores a los Campos para Crear un Proveedor:

ID	Código de Identificación Fiscal(CIF)	Nombre del Proveedor	Fecha de Fundación	Tipo Sociedad	Dirección del Proveedor	Código Postal	Teléfono	Email	Sede Central
2	1	Asus	1989-01-19	Sociedad Anónima (S.A.)	C/ Metalurgia 38, Planta 1, 08038 Barcelona	08038	932 93 81 54	asus@gmail.com	Taipei,Taiwan
3	2	MSI	1986-01-17	Sociedad Anónima (S.A.)	Avinguda Diagonal, 468, 08006 Barcelona	08006	931 845 542	msi@gmail.com	Taipei,Taiwan
42	3	Dell	1984-02-01	Sociedad Anónima (S.A.)	Ribera del Loira, 8, 28042 Madrid	28042	900 827 780	dell@gmail.com	Round Rock, Texas, Estados Unidos
44	4	ACER	1976-08-01	Sociedad Anónima (S.A.)	Carrer del Disseny, 3, 08850 Gavà, Barcelona	08850	(34) 914 - 142 - 414	acer@gmail.com	Taipei,Taiwan

Records per page: 10 ▾ 1-4 of 4

Justo después de crear un proveedor se actualizará automáticamente el modelo de Proveedores.

Si nos dirigimos a la tabla de proveedores y hacemos clic sobre la fila de un proveedor se nos abrirá una página donde podremos ver la información más relevante del proveedor.

CIF: 4

Nombre: ACER
Ciudad: Taipei,Taiwan
C.P: 08850
Telefono: (34) 914 - 142 - 414
Email: acer@gmail.com

Una vez dentro de la **página que explica las características más relevantes de un proveedor** en particular podremos hacer **uso de dos funcionalidades** las cuales son **editar y borrar** para ello tendremos que **clicar en uno de los botones** para **desarrollar una función u otra**.

Si clicamos el **botón de color amarillo** podremos **editar el proveedor**, pero teniendo siempre **en consideración** durante la edición las restricciones y relaciones del modelo Proveedores.

Editar Proveedor:

CIF:
4

Nombre:
ACER

Fecha de Fundacion:
01 / 08 / 1976

Tipo Sociedad:
Sociedad Anonima (S.A.)

Direccion del Proveedor:
Carrer del Disney, 3, 08850 Gava, Barcelona

Codigo Postal:
08850

Telefono:
(34) 914 - 142 - 414

Email:
acer@gmail.com @email.com

Sede Central:
Taipei, Taiwan

Si clicamos el **botón de color rojo** borraremos el proveedor que hemos hecho.

Tanto si **editamos** como **eliminamos** se **actualizará automáticamente** el **modelo de Proveedores**. Si ese **proveedor** estaba **vinculado** previamente a **unos productos** y a **unas transacciones de compra** automáticamente también se **actualizará el modelo de productos y el modelo compras**.

Sección de Clientes

En esta **sección de la aplicación** podremos **observar todos los valores de los clientes que tenemos registrados y** también podremos **crear, editar y eliminar registros** referentes a los **clientes**.

ID	Número de Identificación Fiscal(NIF)	Nombre	Fecha de Registro	Dirección	Teléfono	Email	Código Postal	Localización
3	1	Alvaro	2022-05-06	Plaza España	692100284	al17mora@gmail.com	46960	Aldaya
4	3	Miguel	2022-05-02	Calle Atocha	(45) 646 - 564 - 56	miki@gmail.com	46960	Aldaya
7	4	Angeles	2022-05-02	Calle Ademuz	692100297	angeles@gmail.com	46001	Valencia
35	5	Blas	2022-05-01	Calle del Coscolar	692101284	blas@gmail.com	23001	Jaén

Si clicamos sobre el botón **crear cliente** se nos abrirá una página con un **formulario** que deberemos **rellenar con los valores** que deseamos que **tenga el nuevo cliente** teniendo en cuenta las **restricciones y relaciones** que tiene el **modelo de Clientes del módulo de Administración de Moracorp**. Deberemos **cumplir las restricciones** porque en el caso de que no lo hagamos se **impedirá la creación del nuevo registro**.

Introduce valores a los Campos para Crear un Cliente:

Justo después de **crear un cliente** se **actualizará automáticamente el modelo de Clientes**.

Una vez dentro de la página que explica las características más relevantes de un cliente en particular podremos hacer uso de dos funcionalidades las cuales son editar y borrar para ello tendremos que clicar en uno de los botones para desarrollar una función u otra.

Si clicamos el botón de color amarillo podremos editar el cliente, pero teniendo siempre en consideración durante la edición las restricciones y relaciones del modelo Clientes.

Si clicamos el botón de color rojo borraremos el cliente que hemos hecho.

Tanto si editamos como eliminamos se actualizará automáticamente el modelo de Clientes. Si ese cliente estaba vinculado previamente a unas ventas y a unos pedidos de compra automáticamente también se actualizará el modelo de Ventas y el modelo de Pedidos.

Sección de Pedidos

En esta sección de la aplicación podremos observar todos los valores de los pedidos que tenemos registrados y también podremos crear, editar y eliminar registros referentes a los pedidos. Además, en esta sección se podrá observar en un calendario todos los pedidos de venta como eventos en el mismo, este calendario nos permitirá observar de una forma gráfica y sencilla los plazos de entrega de los pedidos.

Pedidos												
Id	Nº de Pedido	Identificador de la Venta	Distribuidor	NIF del Cliente	Nombre del Cliente	Dirección del Cliente	Teléfono del Cliente	Email del Cliente	Empresa de Logística	Fecha de Expedición	Fecha de Recepción	Estado
3	1	1	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	SEUR	2022-05-03	2022-05-03	En el Almacén
4	2	2	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	UPS	2022-05-02	2022-05-12	En Reparto
5	3	3	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	SEUR	2022-05-02	2022-05-11	En Reparto
6	4	4	Moracorp Electronics	3	Miguel	Calle Atocha	(45) 646-564-56...	miki@gmail.com	UPS	2022-05-18	2022-05-19	Entregado
7	5	5	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	SEUR	2022-05-02	2022-05-11	Entregado
32	6	7	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	DHL	2022-05-23	2022-05-26	Entregado

Si clicamos sobre el botón calendario de pedidos se nos abrirá una página donde podremos observar un calendario con los plazos de entrega de todos los pedidos de venta que ha tramitado la empresa Moracorp Electronics en un determinado plazo de tiempo.

Pedidos :

- Nº de Pedido: 1, Fecha de Expedicion: 2022/05/03, Fecha de Recepcion: 2022/05/03
- Nº de Pedido: 2, Fecha de Expedicion: 2022/05/02, Fecha de Recepcion: 2022/05/11
- Nº de Pedido: 3, Fecha de Expedicion: 2022/05/02, Fecha de Recepcion: 2022/05/11
- Nº de Pedido: 4, Fecha de Expedicion: 2022/05/18, Fecha de Recepcion: 2022/05/19
- Nº de Pedido: 5, Fecha de Expedicion: 2022/05/02, Fecha de Recepcion: 2022/05/11
- Nº de Pedido: 6, Fecha de Expedicion: 2022/05/23, Fecha de Recepcion: 2022/05/26

Si clicamos sobre el **botón crear pedido** se nos abrirá una página con un formulario que deberemos **rellenar con los valores** que deseamos que tenga el nuevo pedido teniendo en cuenta las restricciones y relaciones que tiene el **modelo de Pedidos del módulo de Administración de Moracorp**. Deberemos **cumplir las restricciones** porque en el caso de que no lo hagamos se **impedirá la creación del nuevo registro**.

Pedidos												
Id	Nº de Pedido	Identificador de la Venta	Distribuidor	NIF del Cliente	Nombre del Cliente	Dirección del Cliente	Teléfono del Cliente	Email del Cliente	Empresa de Logística	Fecha de Expedición	Fecha de Recepción	Estado
3	1	1	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	SEUR	2022-05-03	2022-05-03	En el Almacén
4	2	2	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	UPS	2022-05-02	2022-05-12	En Reparto
5	3	3	Moracorp Electronics	4	Angeles	Calle Ademuz	692100297	angeles@gmail.com	SEUR	2022-05-02	2022-05-11	En Reparto
6	4	4	Moracorp Electronics	3	Miguel	Calle Atocha	(45) 646-564-56...	miki@gmail.com	UPS	2022-05-18	2022-05-19	Entregado
7	5	5	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	SEUR	2022-05-02	2022-05-11	Entregado
32	6	7	Moracorp Electronics	1	Alvaro	Plaza España	692100284	al17mora@gmail.com	DHL	2022-05-23	2022-05-26	Entregado
34	7	8	Moracorp Electronics	5	Bias	Calle del Coscollar	692101284	bias@gmail.com	DHL	2022-05-30	2022-05-31	En Reparto

Records per page: 10 - 1-7 of 7

Justo después de **crear un pedido** se **actualizará automáticamente el modelo de Pedidos**.

<input type="checkbox"/>	34	7	8	Moracorp Electronics	5	Bias	Calle del Coscollar	692101284	bias@gmail.com	DHL	30/05/2022	31/05/2022	En Reparto
--------------------------	----	---	---	----------------------	---	------	---------------------	-----------	----------------	-----	------------	------------	------------

Si nos dirigimos a la **tabla de pedidos** y hacemos **clic sobre la fila de un pedido** se nos **abrirá una página** donde podremos **ver de una forma muy detallada toda la información del pedido**. Y incluso **esta página** contará con un **calendario** donde se podrá observar el plazo de entrega que tendrá el pedido.

Una vez dentro de la **página que explica todas las características de un pedido** en particular podremos hacer **uso de dos funcionalidades** las cuales **son editar y borrar** para ello tendremos que **clicar en uno de los botones** para desarrollar una función u otra.

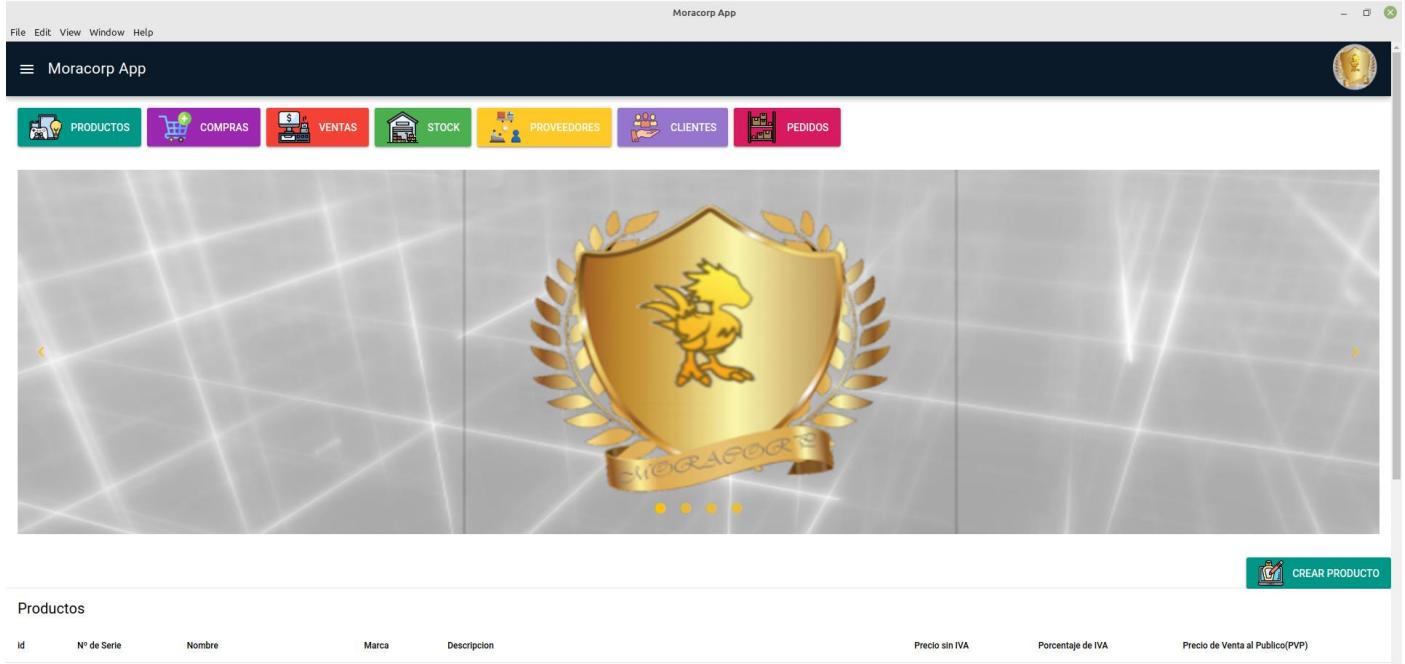
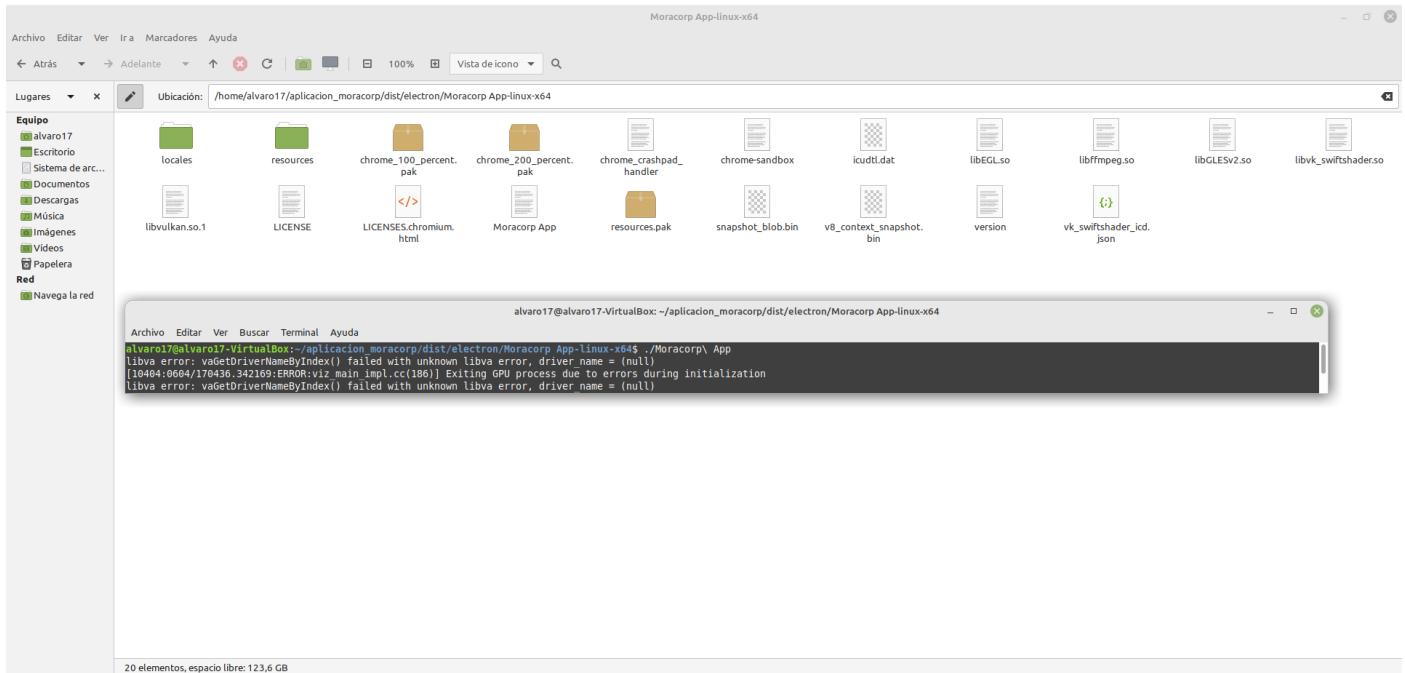
Si clicamos el **botón de color amarillo** podremos **editar el pedido**, pero teniendo siempre en consideración durante la edición las restricciones y relaciones del modelo Pedidos.

Si clicamos el **botón de color rojo** **borraremos el pedido que hemos hecho**.

Tanto si **editamos como eliminamos** se **actualizará automáticamente el modelo de Pedidos**.

Aplicación de Escritorio

Este proyecto de aplicación que he estado desarrollando en el framework front-end Quasar lo he podido exportar o migrar a una aplicación de escritorio real gracias al framework Electron, este framework nos permite crear aplicaciones gráficas de escritorio usando componentes y elementos del lado del cliente y del servidor, que originalmente eran utilizados para el desarrollo de aplicaciones web.

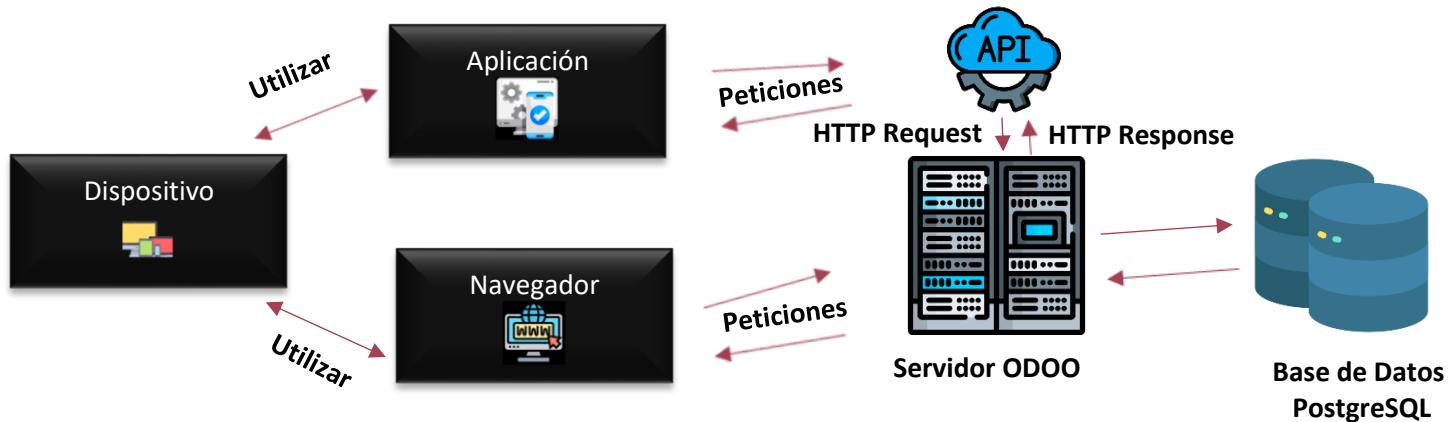


Despliegue

Diagrama de Despliegue

He diseñado un **diagrama de despliegue de mi proyecto**, este **diagrama** nos permitirá **observar de una forma sencilla y grafica la interacción entre los distintos softwares** que he **implementado en este proyecto**.

Este **diagrama** básicamente está **representando como el usuario hace uso de uno de sus dispositivos electrónicos**, para **utilizar el navegador o la aplicación con el fin de acceder al servidor donde está ubicado el ERP**, si hace **uso de la aplicación** hará una serie de peticiones a las **APIS** para poder interactuar con el **Servidor OODO**, si hace **uso del navegador web** hará una serie de peticiones para acceder e interactuar directamente con el **Servidor OODO**. Y dicho **Servidor OODO** estará respaldado por una base de datos **PostgreSQL**, en esta **base de datos** se almacenara toda la **información del ERP**.



Conclusión

A modo de cierre de este proyecto voy a exponer una breve recapitulación sobre los **objetivos que se han cumplido** de esa **idea inicial de proyecto** y de los **conocimientos y valores** que me ha aportado la elaboración de este. Y además comentare una **serie de mejoras** que se podrían **aplicar en este proyecto** en un futuro.

Como se ha podido observar en esta **memoria del proyecto** se han **cumplido en su totalidad todas las ideas** que se habían indicado o expuesto para la **implementación del proyecto de final de grado**. Este proyecto se basa en la implementación de un **ERP** que se **encargará de la gestión y administración de una empresa de electrónica que ofrece compra y venta de sus productos y da soporte sobre esos mismos productos**, a raíz de esa **idea primigenia** comenzó el desarrollo de un **ERP adaptado y configurado** para la **empresa Moracorp Electronics** que **centralizará todos los recursos y servicios básicos** que necesitaría la **compañía para desarrollar la labor**. Una de las ideas que tenía más clara desde un principio es que quería **facilitar la comunicación e interacción entre la empresa y sus clientes y proveedores**, por ello cree un **Servidor de Correos** que permitiera **establecer una comunicación de una forma eficiente, directa y sencilla**. Otra de las ideas que quería implementar era **crear un sitio web personalizado** para la **empresa** que les sirviera como un **escaparate virtual al mundo** donde se pudiera realizar **transacciones y donde se pudiera publicitar la imagen de marca de la empresa**, está la idea la **pude implementar gracias al módulo Sitio Web** del **ERP**, este **módulo** me permitió llevar a cabo la **creación y diseño de la página web**. La **idea principal** por la cual se iba construir este proyecto era la **creación de un módulo personalizado** para la **empresa** que se iba a encargar de la **gestión de los productos y del stock de los mismos, de las compras y ventas de productos, de la tramitación y administración de los pedidos** y también se encargaría de dar **un buen soporte en las comunicaciones y relaciones con los clientes como con los proveedores**, y además este **módulo** permitiría la **tramitación y administración de los documentos** de la **empresa** mediante la **creación de informes**, este **módulo personalizado** sea **podido desarrollar** tal cual lo he explicado gracias a **una gran dedicación y esfuerzo sobre él**. Y la última **idea fundamental** que se quería implementar era la **creación de una aplicación de escritorio** que permitiera **interactuar de una forma sencilla e intuitiva sobre el ERP**,

esta idea se pudo llevar a término y conseguí desarrollar una aplicación eficiente y muy intuitiva que permite observar y interactuar con toda la información del módulo personalizado de Administración Moracorp del ERP desde cualquier dispositivo electrónico.

Este proyecto se podría decir que ha sido un éxito porque se ha podido implementar todo lo que planteé pero no ha sido nada fácil me tuve que superponer a una gran cantidad de problemas, como que no se me actualizara el stock de los productos automáticamente a causa de una falta de experiencia y conocimientos en sustituir métodos fundamentales como crear, editar y borrar en el Servidor OODOO, pero al final pude solventar el problema documentándome y creando esos métodos que sustituían a los métodos fundamentales de OODOO. Otro de los grandes problemas de este proyecto fue la falta de conocimientos con el framework front-end Quasar y también la falta de experiencia con el lenguaje JavaScript mi principal problema era que no sabía cómo interactuar con variables que declaraba en el código HTML y luego tenía que utilizar para insertar información en el lenguaje de programación JavaScript , me acuerdo que me costó horrores insertar la información que cogía de las APIS del Servidor OODOO dentro de componentes como tablas, gráficos y calendarios pero a base de tutoriales y documentación sobre Quasar conseguí hacerlo. Y el último problema importante que tuve fue con las cabeceras HTTP (CORS) que no me permitían editar y eliminar ningún registro desde la aplicación que se conectaba al módulo del Servidor OODOO mediante el uso de una API, solvente este problema gracias a la información que me suministró mi tutor de proyecto.

Todos estos problemas para mi han sido difíciles de solventar y he perdido un buen tiempo con ello, pero mirando el lado bueno de estos problemas es que me han permitido ahondar y expandir mis conocimientos sobre los softwares y herramientas que he utilizado para la elaboración de este proyecto, pero no solo de los problemas se aprende, toda la tarea que he hecho de documentación y análisis para la elaboración del trabajo ha sido fundamental para llevarlo a término. Gracias a este proyecto he podido mejorar mis conocimientos en OODOO y en Docker, también me ha proporcionado experiencia con herramientas de control de versiones como Git, y además he conseguido nuevos conocimientos en un software que no había utilizado nunca como es Quasar. Y incluso he mejorado mi faceta de tester porque he tenido que probar hasta la saciedad los programas que he implementado en busca de fallos o errores.

Para concluir este apartado indicará una serie de mejoras que se podrían hacerse sobre este trabajo en un futuro. Como por ejemplo instalar un proxy que haga de intermediario entre el Servidor OODOO y la aplicación. Otra mejora que se podría hacer en un futuro sería hacer un mejor diseño y decoración de la aplicación. También estaría bien como mejora implementar en el módulo y en la aplicación una sección de soporte técnico. Otra buena funcionalidad que se podría hacer es implementar una funcionalidad que te creará tickets y códigos de barras. Y otra mejora también podría ser hacer uso de más módulos básicos que nos proporciona el Servidor OODOO. Otra mejora que se podría hacer en este trabajo seria la creación de un login donde los usuarios de la aplicación Moracorp pudieran autentificarse para poder acceder al software.

GitHub

He subido la totalidad de mi proyecto a dos repositorios de la plataforma en la nube GitHub. En uno de los repositorios he subido la memoria y todo el contenido relacionado con OODOO como son el tema backend del ERP, la base de datos del ERP y el módulo personalizado de OODOO que he desarrollado. Y en el otro repositorio he subido la aplicación que he desarrollado con el framework front-end Quasar.

Url del repositorio de OODOO:

https://github.com/Alvaro-Mora17/Proyecto_Moracorp_Electronics_OODOO

Url del repositorio de la Aplicación hecha en Quasar:

https://github.com/Alvaro-Mora17/Proyecto_Moracorp_Electronics_Aplicacion_Quasar

Credenciales del Servidor ODOO



Warning, your Odoo database manager is not protected. To secure it, we have generated the following master password for it:

58c8-t2is-y4wg

You can change it below but be sure to remember it, it will be asked for future operations on databases.

Master Password	Invicta17	<input type="button" value="eye"/>
Database Name	Moracorp_Electronics	
Email	al17mora@gmail.com	
Password	Invicta17	<input type="button" value="eye"/>
Phone number	6921002890	
Language	Spanish / Español	<input type="button" value="▼"/>
Country	Spain	<input type="button" value="▼"/>
Demo data	<input type="checkbox"/>	

Create database or restore a database

DockerHub

He subido las imágenes de los contenedores de Docker que he utilizado para la elaboración del proyecto a la plataforma DockerHub.

Url de la imagen del contenedor del Servidor ODOO:

https://hub.docker.com/r/alvaro117mora/moracorp_electronics/tags

Url de la imagen del contenedor del Gestor de Base de Datos PostgreSQL:

https://hub.docker.com/r/alvaro117mora/moracorp_electronics_bd/tags

Glosario

ERP: Enterprise Resource Planning, que significa en castellano **Sistema de Planificación de Recursos Empresariales**, el objetivo de un **ERP** es englobar en un único software todas las herramientas que necesita una empresa para estar operativa en todos los principales departamentos.

ODOO: es un **completo Sistema de Gestión Empresarial (ERP)** de código abierto y sin coste de licencias que cubre las necesidades de la empresa gracias a una amplia gama de herramientas de fácil uso que nos permiten optimizar y rentabilizar los negocios.

PostgreSQL: es un **sistema de base de datos relacional de alta disponibilidad**. Es capaz de funcionar de manera estable en el servidor y, por lo tanto, resulta robusto, una de las principales características que buscan las empresas. Además, es consistente y tolerante a fallos. Es compatible con el modelo relacional, ya que asegura siempre su integridad referencial. Además, este gestor está orientado a objetos, es multiplataforma y es de código abierto.

Python: es un **popular lenguaje de programación interpretado** que puede utilizarse para una gran variedad de aplicaciones. Incluye estructuras de datos de alto nivel, tipificación dinámica, vinculación dinámica y muchas más características que lo hacen tan útil para el desarrollo de aplicaciones complejas como para la creación de scripts o "código cola" que conecta componentes entre sí. Debido a su ubicuidad y a su capacidad para ejecutarse en casi todas las arquitecturas de sistemas, Python es un lenguaje universal que se encuentra en una gran variedad de aplicaciones diferentes. **Python** es un lenguaje de programación interpretado, interactivo y orientado a objetos. Incorpora módulos, excepciones, tipado dinámico, tipos de datos dinámicos de muy alto nivel y clases.

Lenguaje Interpretado: es aquel que el **código fuente se ejecuta directamente**, instrucción a instrucción. Esto quiere decir que el código no pasa por un proceso de compilación, sino que tenemos un programa llamado intérprete que lee la instrucción en tiempo real, y la ejecuta.

Multiparadigma: es el cual soporta más de un **paradigma de programación**. Según lo describe Bjarne Stroustrup, permiten crear “programas usando más de un estilo de programación”. Un lenguaje multiparadigma puede hacer uso de los siguientes paradigmas de la programación orientados a objetos, reflexivos, imperativos y funcionales. El objetivo en el diseño de estos lenguajes es permitir a los programadores utilizar el mejor paradigma para cada trabajo, admitiendo que ninguno resuelve todos los problemas de la forma más fácil y eficiente posible.

Quasar: es un **framework front-end** orientado al desarrollador con componentes **VueJS** para los mejores sitios web receptivos, de alto rendimiento, **PWA**, **SSR**, aplicaciones móviles y de escritorio, todo desde la misma base de código. **Quasar** es un **framework de Javascript** que se basa en **Vue.js**, está optimizado para desarrollar aplicaciones de alto rendimiento en tiempo récord. Es un **framework** de código abierto y ha demostrado ser el más fácil de usar para crear aplicaciones de una sola página (**SPA**), proyectos de renderizado del lado del servidor (**SSR**), aplicaciones web progresivas (**PWA**), así como aplicaciones móviles y de escritorio. Gracias a todos los componentes y herramientas que ofrece **Quasar** listos para usar, los desarrolladores no necesitan bibliotecas de terceros como Bootstrap o Hammer.js. Este **framework** unifica en un mismo archivo el **lenguaje de marcado HTML** y **lenguaje de estilos CSS** con el **lenguaje de programación Javascript**.

Vue.js: es un **framework progresivo de JavaScript** de código abierto para la construcción de interfaces de usuario y aplicaciones de una sola página. Este **framework front-end** hace uso del **lenguaje de marcado HTML**, del **lenguaje de estilos o diseños CSS** y del lenguaje de programación **Javascript** y a su vez es compatible con **WebComponents**. El **Framework Vue.js** es **progresivo** porque es ideal para migrar y adaptar proyectos existentes hechos en otras tecnologías y pasarlo poco a poco a **Vue**. Este **framework** se caracteriza principalmente por ser progresivo y por su sencilla curva de aprendizaje.

Licencia MIT: es una **licencia de software** que acuño el **Instituto Tecnológico de Massachusetts**, esta licencia también se le denomina licencia **X11**. Esta licencia es una Licencia de software libre lo que significa que impone muy pocas limitaciones en la utilización de la misma. La Licencia **MIT** otorga al usuario final del software derechos tales como

copiar, modificar, fusionar, distribuir, etc. Esta licencia es notable por **no tener cláusulas para publicidad y copyright** (La licencia no tiene prohibiciones en el uso del nombre del propietario de los derechos de autor para usos promocionales). La licencia **MIT** permite reutilizar software dentro de Software propietario. Por otro lado, la licencia **MIT** es compatible con muchas licencias copyleft, como la **GNU General Public License**

JavaScript: es un **lenguaje de programación de alto nivel interpretado** que es utilizado para el desarrollo web en el lado del servidor como en el de cliente, aunque este más focalizado dicho lenguaje en el lado del cliente. **JavaScript** es un lenguaje de programación basado en prototipos, multiparadigma, de un solo hilo, es ligero, dinámico, interpretado, compilado, débilmente tipado y con soporte para programación orientada a objetos y también es imperativo y declarativo. Este lenguaje es principalmente utilizado por parte de programadores web para dar respuestas rápidas a las acciones del usuario sin necesidad de enviar la información de lo que ha hecho el usuario al servidor y esperar respuesta de éste (lo que haría más lento los procesos). El código **JavaScript** se carga al mismo tiempo que el código **HTML** en el navegador, y reside en el cliente (computador en el que nos encontramos), por lo que **JavaScript** sigue funcionando incluso aunque se produzca un corte en la conexión a internet (en este caso no podremos seguir navegando hacia otras direcciones web, pero sí podremos ejecutar procesos “locales” en nuestro computador para la página web en que nos encontraremos). También podemos utilizar **JavaScript** para el desarrollo de todo el **back-end** de una aplicación, programando con **Node.js**.

HTML: el **Lenguaje de Marcado de Hipertexto (HTML)** es el código que se utiliza para estructurar y desplegar una página web y sus contenidos. Por ejemplo, sus contenidos podrían ser párrafos, una lista con viñetas, o imágenes y tablas de datos. **HTML** no es un lenguaje de programación, es un lenguaje de marcado que define la estructura del contenido de una página web. **HTML** consiste en una serie de elementos que usarás para encerrar diferentes partes del contenido para que se vean o comporten de una determinada manera.

CSS: es lo que **se denomina lenguaje de hojas de estilo en cascada** y se usa para estilizar y diseñar elementos escritos en un lenguaje de marcado como **HTML**. **CSS** separa el contenido de la representación visual del sitio.

Docker: es un **sistema de virtualización y gestor de contenedores** de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos. El uso de este sistema de virtualización nos proporciona principalmente los siguientes beneficios:

- **Menos recursos, ahorro en costes.** En un mismo host pueden desplegarse más contenedores que máquina virtual (**MV**) tradicionales. Las exigencias en el proceso de inicio y espacio en disco son menores y más rápidas.
- **Gestión de la tecnología de la información (TI)** más fácil, aumento de la productividad de la **tecnología de la información**. La creación de contenedores permite estandarizar los despliegues ya que son entornos repetibles para tareas de desarrollo, prueba y producción. La compatibilidad con todos los sistemas de implantación elimina un valioso tiempo de configuración. Son elementos totalmente portables. Con Docker la implementación se realiza en segundos.
- **Múltiples aplicaciones independientes en un mismo host.** Cada aplicación se ejecuta en su contenedor o clúster de contenedores de forma independiente, sin entrar en conflicto con el resto de las aplicaciones que aloje el host que ejecutarán mediante sus propios contenedores. Esto garantiza un entorno seguro y eficiente.

DockerHub: es un servicio proporcionado por **Docker para buscar y compartir imágenes de contenedores** entre distintos usuarios. Es el repositorio de imágenes de contenedores más grande del mundo con una variedad de fuentes de contenido que incluyen desarrolladores de la comunidad de contenedores, proyectos de código abierto y proveedores de software independientes (**ISV**) que crean y distribuyen su código en contenedores.

Git: es un **sistema de gestión y control de versiones**. Este sistema nos permitirá llevar a cabo la gestión de las distintas versiones de un proyecto. Git presenta **una arquitectura distribuida**, es un ejemplo de **DVCS** (sistema de control de versiones distribuido, por sus siglas en inglés). En lugar de tener un único espacio para todo el historial de versiones del software, como sucede de manera habitual en los sistemas de control de versiones antaño populares, como **CVS**

o **Subversión** (también conocido como SVN), en Git, la copia de trabajo del código de cada desarrollador es también un repositorio que puede albergar el historial completo de todos los cambios.

GitHub: es una **plataforma en la nube** que se utiliza para alojar proyectos utilizando el sistema de control de versiones **Git**. Es una plataforma que puede mantener repositorios de código en almacenamiento basado en la nube para que varios desarrolladores puedan trabajar en un solo proyecto y ver las ediciones de cada uno en tiempo real.

VirtualBox: es un potente software de virtualización **x86** y **AMD64/Intel64** para uso empresarial y doméstico. **VirtualBox** no solo es un producto extremadamente rico en funciones y de alto rendimiento para clientes empresariales, sino que también es la única solución profesional que está disponible gratuitamente como software de código abierto bajo los términos de la licencia pública general **GNU (GPL)**.

Bibliografía

Páginas Webs de las Tecnologías o Plataformas utilizadas para la elaboración del proyecto:

<https://www.docker.com/>

<https://hub.docker.com/>

https://www.odoo.com/es_ES

<https://www.postgresql.org/>

<https://www.python.org/>

<https://quasar.dev/>

<https://git-scm.com/>

<https://github.com/>

Urls que te redireccionan a ciertas secciones de la Página Web de la librería Bootstrap donde se explica como instalarla (esta librería la he utilizado para aplicar estilos a los informes dentro del Servidor OODOO. Esta librería no hace falta instalarla para utilizarla en OODOO porque ya cuenta el propio ERP con ella por defecto) y como implementarla.

<https://getbootstrap.com/>

<https://getbootstrap.com/docs/5.1/content/tables/>

Urls que te redireccionan a ciertas secciones de la Página Web de la librería Axios donde se explica como instalarla y implementarla.

<https://axios-http.com/es/docs/intro>

https://axios-http.com/es/docs/api_intro

Video Tutorial donde se explica al detalle cómo hacer consultas con la librería Axios desde el framework front-end Quasar.

<https://www.youtube.com/watch?v=HOv-TTCIcp0>

Urls que te redireccionan a ciertas secciones de la Página Web de la librería Apexcharts donde se explica como instalarla y implementarla, esta librería nos brinda cientos de modelos gráficos que se pueden utilizar como base para crear nuestros propios gráficos.

<https://apexcharts.com/docs/installation/>

<https://apexcharts.com/>

<https://apexcharts.com/vue-chart-demos/column-charts/>

Video Tutorial donde se explica al detalle como implementar la librería Apexcharts en un proyecto hecho en Quasar.

<https://www.youtube.com/watch?v=v65vjuetRNg>

