



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

CAMPUS D'ALCOI

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Politécnica Superior de Alcoy

Análisis y selección de un ERP para una PYME de
servicios profesionales de interiorismo

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: López Del Pino, Miguel

Tutor/a: Llorca Alcón, Manuel

CURSO ACADÉMICO: 2022/2023

Resumen

La evolución de los sistemas de gestión en las últimas décadas ha influenciado a las empresas modernizando sus hábitos de trabajo. Esto ha llegado a empresas de gran tamaño hasta a las PYMES y microempresas, que han tenido que adaptar la forma de gestionar su negocio para lograr su supervivencia en el mundo empresarial.

Con este trabajo se pretende dar una solución Opensource, de fácil manejo e implementación, permitiendo la adaptación más conforme al modelo de negocio de la empresa. Para ello se realizarán en distintas fases, empezando por analizar las necesidades del cliente y una vez realizado se hará la selección del software más apropiado. Tras esto, se implantará el ERP y una vez implantado se hará un desarrollo a medida personalizando el código fuente según las necesidades previamente identificadas, con la realización de pruebas pertinentes.

Palabras Clave

ERP – Sistemas de Gestión – Odoo – Desarrollo a Medida – Vistas y Modelos

Summary

The evolution of management systems in recent decades has influenced companies and modernised their working habits. This has reached from large companies to SMEs and micro-enterprises, which have had to adapt the way they manage their business in order to survive in the business world.

The aim of this work is to provide an Opensource solution, easy to use and implement, allowing the most appropriate adaptation to the company's business model. This will be carried out in different phases, starting by analysing the client's needs and once this has been done, the most appropriate software will be selected. After this, the ERP will be implemented and once it has been implemented, a customised development will be carried out, personalising the source code according to the previously identified needs, with the relevant tests being carried out.

Keywords

ERP - Management Systems - Odoo - Custom Development - Views and Models

Índice

| | |
|--|----|
| 1.1 Contexto | 6 |
| 1.2 Motivación | 7 |
| 1.3 Objetivos del Trabajo académico..... | 7 |
| 1.4 Estructura del TFG | 8 |
| 2. LOS SISTEMAS DE INFORMACIÓN EN LA EMPRESA | 9 |
| 2.1 El dato..... | 9 |
| 2.2 Información..... | 9 |
| 2.3 Conocimiento..... | 9 |
| 2.4 Sistema de Información (SI) | 11 |
| 2.5 Componentes de un SI | 11 |
| 2.6 Características de un SI | 12 |
| 2.7 Ventajas y Desventajas de un SI..... | 13 |
| 2.7.1 Ventajas:..... | 13 |
| 2.7.2 Problemas Asociados | 14 |
| 2.8 Procesos de un SI..... | 14 |
| 2.9 Alineamiento estratégico..... | 15 |
| 2.10 Tipos de SI en las empresas..... | 16 |
| 2.10.1 Perspectiva Funcional | 16 |
| 2.10.1 Perspectiva Usuario..... | 17 |
| 3. ENTERPRISE RESOURCE PLANNING..... | 18 |
| 3.1 Qué es un ERP | 18 |
| 3.2 Principales Características del ERP | 19 |
| 3.3 Ventajas y Desventajas..... | 20 |
| 3.3.1 Ventajas:..... | 20 |
| 3.3.1 Desventajas: | 20 |
| 3.4 Tipos de ERP | 21 |
| 3.4.1 Diseño:..... | 21 |
| 3.4.2 Alojamiento:..... | 22 |
| 3.4.3 Solución:..... | 22 |
| 3.4.4 Desarrollo: | 23 |
| 4. La Empresa | 24 |
| 4.1 Prólogo | 24 |
| 4.2 La Empresa | 25 |
| 4.3 Situación Previa..... | 25 |

| | |
|---|----|
| 4.4 Procesos de la empresa..... | 26 |
| 4.4.1. Gestión de Clientes | 26 |
| 4.4.2 Contabilidad y Facturación..... | 27 |
| 4.4.3 Gestión de Compras | 27 |
| 4.4.4 Gestión de Ventas | 27 |
| 4.5 Resumen Necesidades | 28 |
| 5. Selección del ERP y Planificación del Proyecto | 28 |
| 5.1 Selección del ERP..... | 28 |
| 5.2 Comparativa y Selección. | 29 |
| 5.2.1 Apache OFBiz..... | 29 |
| 5.2.2 Adempiere..... | 30 |
| 5.2.3 Odoo..... | 31 |
| 5.3 Análisis y Selección..... | 32 |
| 5.4 Planificación y Estrategia..... | 34 |
| 5.5 Coste del Proyecto. | 36 |
| 6. Instalación de Odoo | 37 |
| 6.1 Instalación: | 37 |
| 6.2 Configuración Inicial:..... | 42 |
| 6.3 Instalación de Módulos: | 44 |
| 6.3.1 – Instalación Módulo de Ventas:..... | 44 |
| 6.3.2 – Instalación Módulo de Compras: | 45 |
| 6.3.3 – Instalación Módulo de CRM: | 46 |
| 6.3.4 – Instalación Módulo de Inventario: | 47 |
| 7. Personalización Odoo..... | 48 |
| 7.1 Facturación:..... | 48 |
| RESULTADO FINAL..... | 71 |
| 7.2 Configuración Textil: | 72 |
| 8. Conclusiones..... | 83 |
| 8.1 Conclusión: | 83 |
| 8.2 Líneas Futuras: | 84 |
| 9. Referencias y Bibliografía | 86 |

Tabla de Ilustraciones

| | |
|--|----|
| <i>Ilustración 1: Ciclo del Dato</i> | 10 |
| <i>Ilustración 2: Alineamiento estratégico</i> | 15 |
| <i>Ilustración 3: Perspectiva de usuario</i> | 18 |
| <i>Ilustración 4: Matriz ERP</i> | 24 |
| <i>Ilustración 5: Apache OFBiz</i> | 30 |
| <i>Ilustración 6: ADempiere</i> | 31 |
| <i>Ilustración 7: Odoo</i> | 32 |
| <i>Ilustración 8: Diagrama de Gantt</i> | 35 |
| <i>Ilustración 9: Descarga de Odoo</i> | 37 |
| <i>Ilustración 10: Versión de Odoo</i> | 38 |
| <i>Ilustración 11: Idioma de instalación</i> | 38 |
| <i>Ilustración 12: Tipo de Instalación</i> | 39 |
| <i>Ilustración 13: Conexión PostgreSQL</i> | 39 |
| <i>Ilustración 14: Ruta directorio</i> | 40 |
| <i>Ilustración 15: Creación BBDD</i> | 41 |
| <i>Ilustración 16: Acceso BBDD</i> | 41 |
| <i>Ilustración 17: Listado de Módulos</i> | 42 |
| <i>Ilustración 18: Ajustes Odoo</i> | 42 |
| <i>Ilustración 19: Usuarios y Compañías</i> | 43 |
| <i>Ilustración 20: My Company</i> | 43 |
| <i>Ilustración 21: Creación Empresa</i> | 43 |
| <i>Ilustración 22: Módulo de Ventas</i> | 44 |
| <i>Ilustración 23: Despliegue módulo ventas</i> | 44 |
| <i>Ilustración 24: Módulo de ventas</i> | 45 |
| <i>Ilustración 25: Módulo de compras</i> | 45 |
| <i>Ilustración 26: Despliegue módulo compras</i> | 45 |
| <i>Ilustración 27: Módulo compras</i> | 46 |
| <i>Ilustración 28: Módulo de CRM</i> | 46 |
| <i>Ilustración 29: Despliegue módulo CRM</i> | 46 |
| <i>Ilustración 30: Módulo CRM</i> | 47 |
| <i>Ilustración 31: Módulo de Inventario</i> | 47 |
| <i>Ilustración 32: Despliegue módulo inventario</i> | 47 |
| <i>Ilustración 33: Módulo de inventario</i> | 48 |
| <i>Ilustración 34: Factura Odoo</i> | 48 |
| <i>Ilustración 35: Modelo Presupuesto Odoo</i> | 50 |
| <i>Ilustración 36: Modelo y Plantilla Odoo</i> | 50 |
| <i>Ilustración 37: Vista Modelo y Plantilla Odoo</i> | 51 |
| <i>Ilustración 38: Vista QWEB</i> | 51 |
| <i>Ilustración 39: Vista Modelo y Plantilla Odoo</i> | 52 |
| <i>Ilustración 40: Vista "sale_report_template.xml"</i> | 52 |
| <i>Ilustración 41: Vista 2 "sale_report_template.xml"</i> | 53 |
| <i>Ilustración 42: Report "pedido_de_venta_disenalmeria"</i> | 53 |
| <i>Ilustración 43: Llamada a reporte</i> | 54 |
| <i>Ilustración 44: Creación reporte Odoo</i> | 54 |
| <i>Ilustración 45: Creación reporte Odoo_2</i> | 55 |
| <i>Ilustración 46: Creación reporte Odoo_3</i> | 55 |
| <i>Ilustración 47: Vista Botón</i> | 56 |
| <i>Ilustración 48: Modelo Botón</i> | 56 |

| | |
|--|-----------|
| <i>Ilustración 49: Botón Odoo.....</i> | <i>57</i> |
| <i>Ilustración 50: Reporte vacío botón.....</i> | <i>57</i> |
| <i>Ilustración 51: Vista "report_custom_saleorder".....</i> | <i>58</i> |
| <i>Ilustración 52: Vista formato de papel "disenalmeria".....</i> | <i>59</i> |
| <i>Ilustración 53: "Limites de cabeceras".....</i> | <i>59</i> |
| <i>Ilustración 54: Limites de cabeceras_1.....</i> | <i>60</i> |
| <i>Ilustración 55: Limites de cabeceras_2.....</i> | <i>60</i> |
| <i>Ilustración 56: Boceto Reporte.....</i> | <i>61</i> |
| <i>Ilustración 57: Diseño Cabecera.....</i> | <i>61</i> |
| <i>Ilustración 58: Definir Cabecera.....</i> | <i>62</i> |
| <i>Ilustración 59: Definir Cabecera_2.....</i> | <i>62</i> |
| <i>Ilustración 60: Resultado Cabecera.....</i> | <i>62</i> |
| <i>Ilustración 61: Boceto Reporte_2.....</i> | <i>63</i> |
| <i>Ilustración 62: Configuración "Página Reporte".....</i> | <i>63</i> |
| <i>Ilustración 63: Configuración "Página Reporte" 2.....</i> | <i>64</i> |
| <i>Ilustración 64: Resultado tablas de datos.....</i> | <i>64</i> |
| <i>Ilustración 65: Boceto Reporte_2.....</i> | <i>65</i> |
| <i>Ilustración 66: Tabla Productos_1.....</i> | <i>65</i> |
| <i>Ilustración 67: Tabla Productos_2.....</i> | <i>66</i> |
| <i>Ilustración 68: Resultado Tabla Productos.....</i> | <i>67</i> |
| <i>Ilustración 69: Boceto Reporte_3.....</i> | <i>67</i> |
| <i>Ilustración 70: Configuración Tabla Totales_1.....</i> | <i>68</i> |
| <i>Ilustración 71: Configuración Tabla Totales_1.....</i> | <i>68</i> |
| <i>Ilustración 72: Resultado Tabla Totales.....</i> | <i>69</i> |
| <i>Ilustración 73: Boceto Reporte_4.....</i> | <i>69</i> |
| <i>Ilustración 74: Diseño Pie de Página.....</i> | <i>70</i> |
| <i>Ilustración 75: Configuración Pie de Página.....</i> | <i>70</i> |
| <i>Ilustración 76: Configuración Pie de Página_2.....</i> | <i>70</i> |
| <i>Ilustración 77: Resultado Reporte Final.....</i> | <i>71</i> |
| <i>Ilustración 78: Configuración parámetros de medida.....</i> | <i>73</i> |
| <i>Ilustración 79: Configuración de vista de parámetros de medida.....</i> | <i>74</i> |
| <i>Ilustración 80: Producto Tela Jacquard.....</i> | <i>75</i> |
| <i>Ilustración 81: Configuración parámetro proveedor.....</i> | <i>75</i> |
| <i>Ilustración 82: Configuración parámetro proveedor_2.....</i> | <i>76</i> |
| <i>Ilustración 83: Configuración parámetro proveedor_3.....</i> | <i>77</i> |
| <i>Ilustración 84: Configuración parámetro proveedor_4.....</i> | <i>78</i> |
| <i>Ilustración 85: Configuración Tela Jacquard.....</i> | <i>78</i> |
| <i>Ilustración 86: Configuración Tela Jacquard_2.....</i> | <i>79</i> |
| <i>Ilustración 87: Configuración Tela Jacquard_3.....</i> | <i>79</i> |
| <i>Ilustración 88: Configuración Tela Jacquard_4.....</i> | <i>80</i> |
| <i>Ilustración 89: Configuración detalle de factura.....</i> | <i>81</i> |
| <i>Ilustración 90: Resultado configuración detalle de factura.....</i> | <i>82</i> |

1.1 Contexto

El término ERP (Enterprise Resource Planning) se utilizó por primera vez en la década de 1990. Se centraba en funciones empresariales como la producción, finanzas y contabilidad, recursos humanos, gestión de proyectos, etc. Estaba compuesto por sistemas modulares con características avanzadas.

Con el comienzo del nuevo siglo los ERP se popularizaron. A partir del año 2000 encontramos nuevas funcionalidades como la gestión de la cadena de suministro, la gestión de las relaciones con los clientes (CRM) y la inteligencia de negocio.

A partir del 2005 la tendencia se ha orientado hacia soluciones de software en la nube y se ha alejado de los modelos tradicionales de instalación en los servidores del cliente. Las soluciones de software Cloud ERP proporcionaron funcionalidades comparables a las del ERP local a un coste mucho menor.

Desde ese momento se produce un punto de inflexión en el que ya no solo son utilizados por grandes empresas, sino que se extiende su uso a las PYMES.

Pero ¿Cuántas PYMES cuentan con un ERP? ¿De verdad son ERP o se confunde aplicaciones Ofimáticas con ERP?

Tras este contexto, toda organización, independientemente de su tamaño, debería tener en cuenta un SI que le permita interrelacionar los procesos de la empresa y así contar con un recurso estratégico con el que poder hacer frente a un entorno globalizado y cada vez más interconectado, donde el uso de las TI y SI son claves que marcan el éxito o el fracaso.

1.2 Motivación

Tras realizar las prácticas en empresa y haber cursado dos asignaturas que considero muy importantes de mi mención de Sistemas de Información, considero que las empresas que cuentan con herramientas tecnológicas, como los ERP, tienen una gran ventaja competitiva en el mercado, que le permite crecer a mayor velocidad que empresas que no cuentan con un SI.

La motivación de este Trabajo de Fin de Grado surgió a partir de la empresa familiar de un amigo, una PYME que se dedica al Diseño, tanto Interior como Exterior de viviendas, oficinas, locales y otros tipos de negocios.

Esta empresa no cuenta con un Sistema de Información plenamente integrado, y me pareció una buena oportunidad para que la empresa pudiera dar el salto tecnológico y para mi desarrollo como Ingeniero Informático.

1.3 Objetivos del Trabajo académico.

- Desarrollar los conceptos: Dato, Información, Conocimiento, SI y ERP
- Contextualizar los SI en la Empresa
- ERP frente aplicaciones Ofimáticas
- Contexto de la Empresa. Describir sus procesos de negocios y necesidades.
- Explicación del ERP escogido, herramientas necesarias y procesos tras la implantación del ERP.
- Instalación y adaptación del ERP.

1.4 Estructura del TFG

Este TFG está estructurado de forma que:

- Parte 2. Concepto de SI y sus componentes.
- Parte 3. Qué es un ERP, tipos y comparativa.
- Parte 4. La empresa, contexto, problemática, sus procesos de negocio y necesidades previas a la implantación del ERP
- Parte 5. Descripción del ERP elegido, criterios de elección, procesos de negocio y planificación de su implantación.
- Parte 6. Instalación y configuración del ERP.
- Parte 7. Personalización de Odoo.
- Parte 8. Ejemplo de proceso de negocio del ERP y explicación.

2. LOS SISTEMAS DE INFORMACIÓN EN LA EMPRESA

En este apartado explicaremos los conceptos básicos para entender los Sistemas de Información (SI) y los diferentes tipos que pueden existir. Para ello tendremos que diferenciar 3 premisas principales: Dato, Información y Conocimiento, y entender la diferencia entre SI y TI.

2.1 El dato.

En el ámbito de la informática, lo entendemos como la materia prima, que se puede almacenar en registros en distintas formas, como caracteres numéricos o palabras. Por sí solos no conducen a ningún tipo de información y no sirve de nada.

Por ejemplo, tenemos dos elementos, “4150DBJ” y “WAUZZ00897”

2.2 Información.

La información es un conjunto de datos organizados de tal modo si se contextualizan, adquieren un valor adicional. Para ello deberán ser procesados de tal forma que resulten útiles para quien trata y recibe la información.

Con el ejemplo anterior, podemos imaginarnos que “4150DBJ” es la matrícula de un Coche y “WAUZZ00897” es su número de bastidor.

2.3 Conocimiento.

Es el recurso que posibilita la conversión de la información en decisiones y acciones. Quien recibe y trata la información, puede convertir esta información en conocimiento, a través de su experiencia y conocimiento. Con todo ello, podemos tomar decisiones en base a lo mencionado anteriormente.

Con el ejemplo anterior, sabiendo el número de la matrícula o su bastidor, un empleado de recambios puede saber y diferenciar las piezas necesarias para tu coche.

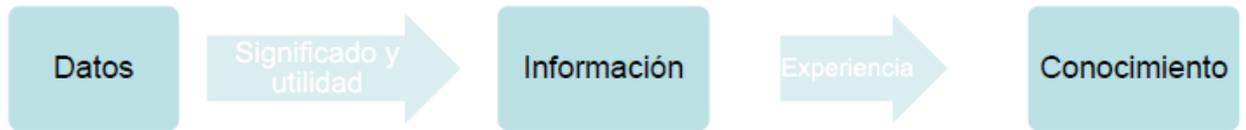


Ilustración 1: Ciclo del Dato

Fuente: Material UPV

Para que este proceso de conversión del dato al conocimiento es necesario que la información cumpla con una serie de cualidades:

- *Exacta*: La información debe ser precisa y libre de errores.
- *Completa*: Debe contener los hechos importantes para la persona que lo va a emplear.
- *Económica*: El beneficio debe ser mayor al coste de obtenerla.
- *Confiable*: Depende de diversos factores, como el método de recolección y tratamiento de datos o la fuente que los genera.
- *Pertinente*: Debe ser realmente importante para la toma de decisiones.
- *Oportuna*: Se debe de entregar en tiempo y forma para poder emplearla correctamente.
- *Simple*: No ha de ser excesivamente compleja.
- *Accesible*: Debe ser de fácil acceso para usuarios autorizados.
- *Verificable*: Ha de poder ser contrastada y comprobada en todo momento.
- *Segura*: Debe de estar protegida contra el acceso de usuarios no autorizados.

2.4 Sistema de Información (SI)

Según la DPEJ se trata de: “*Conjunto de elementos informatizados de acuerdo con un almacenamiento estructurado, que permite el acceso, las relaciones entre sus elementos, la puesta en común y la interoperabilidad con otros sistemas.*” (Real Academia Española, 2011)

Teniendo en cuenta esta definición, se puede entender que un sistema de información es un conjunto de elementos que están organizados y estructurados para que los elementos que lo conforman (datos e información) se pongan en común para un objetivo común. Este objetivo puede ser para ayuda a la toma de decisiones o análisis de interés de la empresa.

Nos podemos encontrar que, junto con definiciones de SI, aparezca el término de TI también, y conviene diferenciarlos entre ellos.

Podemos decir que las TI es el conjunto de herramientas Hardware y Software que emplea el SI para hacer su cometido.

Con todo lo mencionado anteriormente, podemos resumir que un Sistema de Información empresarial es un conjunto de recursos técnicos, humanos y económicos que están interrelacionados y organizados con el objetivo de satisfacer las necesidades de información de una organización empresarial para distintas utilidades como la gestión, analizar problemas, visualizar temas complejos y adopción de decisiones.

2.5 Componentes de un SI

Un SI está formado por distintos elementos que lo conforman y que en su conjunto hace posible su función.

- **Información:** Es la base del SI para su funcionamiento y poder así realizar sus procesos de gestión y tratamiento de la información.
- **Personas:** Son todos los usuarios que interactúan con el SI.
- **Procesos:** Son las prácticas de trabajo habituales de la empresa de los distintos usuarios del SI.

- **TI:** Son las herramientas tanto Hardware como Software que emplean los sistemas informáticos. Por ejemplo, discos duros o sistemas operativos.

También será necesario tener en cuenta factores como:

- **Límites:** Frontera entre lo que es propio del sistema y lo que constituye su entorno.
- **Entorno:** Son todos aquellos elementos exteriores que pueden influir en el Sistema.

2.6 Características de un SI

Los SI, a parte de los objetivos que hemos expuesto anteriormente, tienen también unas características que lo definen para su correcto uso, que son:

- **Fiable:** Facilita la información de calidad y sin errores de transmisión.
- **Relevante:** La información proporcionada es de interés para el usuario que la requiere.
- **Selectivo:** Suministra la información necesaria para el objetivo asignado.
- **Oportuno:** La información se entrega en el momento necesario.
- **Flexible:** El SI permite que la información pueda ser modificada de una forma sencilla para adaptarse a las necesidades cambiantes de la organización.

2.7 Ventajas y Desventajas de un SI.

La implantación de un SI puede ofrecer una serie de ventajas y problemas asociados:

2.7.1 Ventajas:

- **Reducción de costes:** Un SI bien diseñado y eficaz permitirá que el SI pueda asumir más trabajo (Análisis y comparación de datos, resultados, información, etc.) y con mayor rapidez que si las tareas fuesen realizadas de forma estándar por un ser humano.
- **Logra ventajas competitivas:** Gracias al SI se pueden obtener altas rentabilidades respecto de competidores, como reducción de tiempo para analizar un mercado nuevo.
- **Control más efectivo sobre los procesos de Negocio:** Gracias a la monitorización, los procesos de negocio están soportados por el SI, ofreciendo información detallada y ordenada.
- **Apoyo para análisis y estudio de un problema:** Con la información que soporta el SI pueden identificarse problemas y errores internos.
- **Planes Estratégicos:** Ayuda a desarrollar una óptima estrategia operativa permitiendo una planificación más detallada de las principales tareas internas del negocio.
- **Suministrar información a los distintos niveles:** La información será accesible en tiempo y forma que se desee, según quien lo pide, siempre y cuando el SI esté actualizado en tiempo real, siendo de utilidad pudiendo ayudar en la toma de decisiones.
- **Relación con clientes:** Si el SI cuenta con un sistema CRM permitirá fidelizar clientes y fortalecer relaciones

2.7.2 Problemas Asociados

Implantar un SI no es tarea fácil, y pueden ocurrir ciertos problemas:

- **Problemas de Implantación:** Pueden surgir errores de carácter Hardware o Software si no se ha hecho previamente un análisis adecuado.
- **Resistencia al cambio:** Puesto que implantar un SI supone un cambio en los procesos habituales de la empresa, es posible que los usuarios muestren una resistencia al cambio.
- **Tiempo de transición:** Puesto que cada SI es diferente y requiere adaptarse según la empresa, el tiempo de implantación no suele ser corto y requiere de la implicación de todas las partes de la empresa.

2.8 Procesos de un SI.

Hemos visto algunas de las características de los SI y su finalidad, pero conviene saber cómo funciona el SI y sus procesos más importantes.

- **Captación de Información:** Es el primer paso de todos, que puede realizarse manualmente (Información insertada por el usuario) o automática.
- **Procesamiento:** La información se procesa, analiza y trata para alcanzar los objetivos que el propio sistema ha establecido, como toma de decisiones, análisis de problemas internos, etc.
- **Almacenamiento:** En este paso se almacena todo lo realizado anteriormente para que se pueda usar cuando sea demandado. Se almacena y se organiza en un repositorio que puede ser físico o en la nube. Además, allí donde esté almacenada, podrá ser actualizada si así lo requiere.
- **Salida:** Es el paso final donde el usuario que solicita la información que requiere, el SI la procesa y se la ofrece. Según el nivel de usuario que lo solicite, la información podrá ser más o menos detallada.

2.9 Alineamiento estratégico.

Para poder tener una implantación correcta de un SI en nuestra empresa, será necesario una planificación previa de la implantación con una estrategia y metodología que nos permita analizar los puntos más importantes de nuestra empresa para poder seleccionar y adaptar el SI.

Además de ello, será necesario que se documente todo lo realizado y procesado de dicho análisis.

Para que se realice un alineamiento estratégico correcto, se requieren de 4 fases:

- **Conocer la estrategia del negocio:** Conocer los objetivos, contexto y procesos de negocio de la empresa.
- **Balance de Situación:** Analizar la situación de los SI y TI existentes por la empresa y lo que supone en coste, tiempo, productividad, etc.
- **Definición y Formulación de la estrategia:** Partiendo de la situación actual, se debe definir las actividades requeridas para lograr el escenario objetivo y tener una idea clara del beneficio tangible e intangible a obtener derivado de la estrategia.
- **Plan de implantación:** Establecer etapas del proceso, recursos necesarios y gestión del cambio.

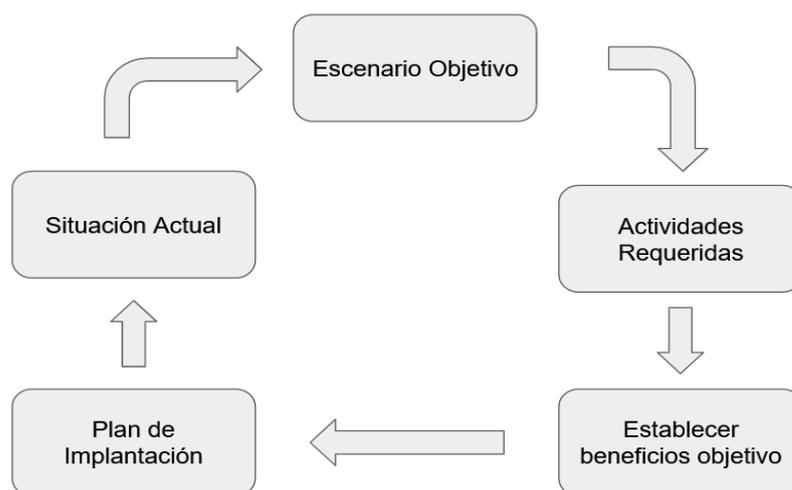


Ilustración 2: Alineamiento estratégico

Fuente: Material UPV

2.10 Tipos de SI en las empresas.

Existen diversos tipos de SI que apoyan a los distintos procesos de negocio de una empresa. Para clasificarlos, principalmente existen 2 perspectivas:

- **Funcional:** Clasifica los sistemas por las principales funciones empresariales.
- **Usuario:** Clasifica los sistemas en función de los principales usuarios o grupos de organización a los que da servicio. Cada nivel de la empresa requiere de necesidades distintas.

2.10.1 Perspectiva Funcional

Podemos diferenciar distintos sistemas bajo esta perspectiva, que son:

- **Sistemas de Ventas y Marketing:** Encargado de actividades de publicidad y venta de la empresa. Se encarga de actividades como la toma de pedidos, registro de ventas o identificación de clientes.
- **Sistemas de Planificación y Producción:** Se encarga de los procesos de Planificación y producción de la empresa, realizando actividades como la planificación de materiales, recursos y mano de obra, establecer objetivos de producción, etc.
- **Sistemas Financieros y Contables:** Realiza las actividades económicas de la empresa, encargándose de actividades como la gestión de activos, rendimiento de inversiones, administrar los registros financieros y sacar estadísticas.
- **Sistemas de Recursos Humanos:** Es el encargado de realizar los procesos de gestión del departamento de Recursos Humanos de la empresa, con funciones como ayudar a la alta dirección a identificar requerimientos de RRHH, supervisión de empleados y mantener la fuerza de trabajo de la empresa.

- **Sistemas logísticos:**

El resto de los sistemas anteriormente explicados, necesitan de la presencia de este último, encargado de tareas que están pensadas más con logística externa que interna, como el envío de productos o estimación de precio y plazo de envíos.

2.10.1 Perspectiva Usuario

Desde esta perspectiva, se clasifican los sistemas en función de los principales grupos de la organización a los que dan servicio y al tipo de decisiones al que van dirigidas:

- **Sistema de Procesamiento de Transacciones (TPS):**

Suele ser de los primeros sistemas que se implantan en la organización y que nutre al resto de sistemas ya que se encarga de ejecutar y registrar las transacciones ordinarias de una empresa, automatización de tareas y procesos operativos, como por ejemplo el sistema de procesamiento de pedidos.

- **Sistemas de información Gerencial (MIS):**

Proporciona a los gestores intermedios informes de rendimiento de la empresa con la finalidad de controlarla y ayudar en la predicción de rendimientos futuros, como, por ejemplo, datos de ventas.

- **Sistema de apoyo a la toma de decisiones (DSS):**

Este sistema necesita de los dos anteriores sistemas, pues es el que se encarga de ayudar a la gerencia intermedia a tomar decisiones poco habituales relacionadas con problemas específicos, como, por ejemplo, como se vería afectado el sistema de producción si se decide sacar un nuevo producto.

- **Sistema de apoyo/información a ejecutivos (EIS):**

Presenta información valiosa, junto con gráficos, que ayuda a la alta dirección a tomar decisiones no rutinarias que requieren juicio, evaluación y comprensión, ya que se tratan de decisiones que no existen procedimientos preestablecidos.

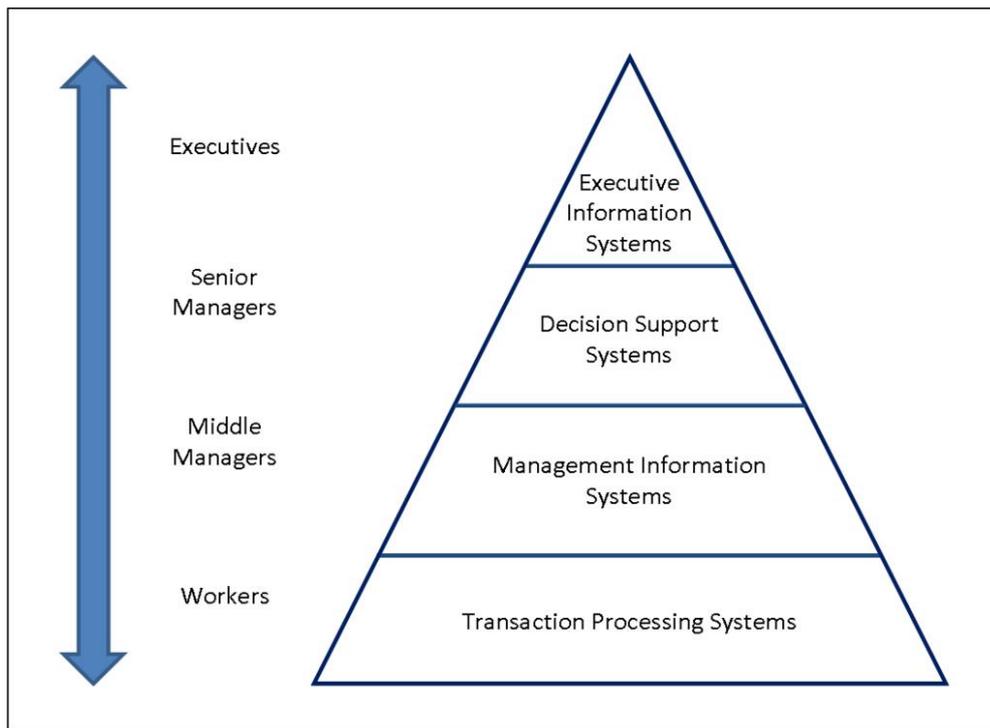


Ilustración 3: Perspectiva de usuario

Fuente: https://es.wikipedia.org/wiki/Sistema_de_informaci%C3%B3n

3. ENTERPRISE RESOURCE PLANNING

3.1 Qué es un ERP

ERP (Enterprise Resource Planning) son las siglas que se utilizan para hablar de los sistemas de planificación de recursos de la empresa sea cual sea su tamaño.

Es un software de gestión global que integra y cubre todas las facetas del negocio, incluyendo planificación, producción, ventas y contabilidad. Desde la misma aplicación se obtiene toda la información que se necesite.

Gracias al ERP, es posible la ejecución y automatización de los procesos de negocio de todas las áreas funcionales de un modo coordinado.

Kumar y Hillegersbeg (2000) describen a los sistemas ERP como: "Paquetes de sistemas de información configurables que integran información y procesos de negocio basados en la información dentro y entre áreas funcionales en una organización" (Kumar y Hillegersbeg,2000).

Orton y Marlene (2004) define a los sistemas ERP como:" Sistema que permite coleccionar y consolidar información a través de la empresa."

Microsoft define los ERP como “Software de gestión de procesos empresariales que administra e integra las actividades de finanzas, cadena de suministro, operaciones, comercio, creación de informes, fabricación y recursos humanos de la empresa.” (Microsoft,2022)

Después de haber visto las diferentes definiciones de ERP, podemos concluir en que el ERP es un *Sistema de información* dividido en distintos módulos según los procesos de negocio de la empresa, donde toda la información generada queda reservada en el propio sistema, y que tiene por finalidad aprovechar la información y los recursos para generar ventajas estratégicas y competitivas de la empresa.

3.2 Principales Características del ERP

Conocer las características de un ERP supone conocer las cualidades que nos podrán ser de utilidad para nuestra empresa.

- **Automatización de Procesos:** El ERP nos permite realizar diferentes funciones de distintas áreas de nuestro negocio de forma automática, desde la cadena de suministro hasta la entrega del producto, consiguiendo mayor eficiencia en la empresa pues la mayoría de estos procesos son operaciones rutinarias que se pueden delegar en esta herramienta.
- **Integración:** Toda la información de cada uno de los departamentos queda centralizada y cohesionada con el resto en un Sistema central, permitiendo que todos los departamentos trabajen juntos. Además, el ERP puede integrarse con otros programas.
- **Modularidad:** Es una de las características más importantes, pues de esta manera es posible diseñar un ERP según las necesidades de la empresa, permitiendo mayor adaptabilidad a la hora de perdurar en el tiempo. Esta modularidad permite que el ERP se pueda diseñar en ‘bloques’.

- **Consistencia:** Otra de las características principales es que elimina la duplicidad de la información, además de que permite realizar cambios de forma segura.

3.3 Ventajas y Desventajas

Incorporar un ERP en tu empresa es un proceso que puede traer ventajas y desventajas en el mismo. Conviene tenerlo en cuenta en el Plan estratégico para así minimizar los problemas que puedan ocurrir.

3.3.1 Ventajas:

Aparte de las ventajas que hemos podido conocer de los sistemas ERP por sus características, también permite una serie de ventajas como:

- **Optimización de Costes:** Aumenta la eficiencia al acceder a la información más fácilmente o por reducción de personal.
- **Mejora de Procesos:** Incorpora las mejores prácticas de los procesos y desde un único sistema.
- **Mejora las relaciones empresariales:** Gracias a que se centraliza en un sistema, permite mejor comunicación entre trabajadores de la propia empresa, proveedores, clientes, socios logísticos, etc.
- **Control y gestión:** Al estar todo integrado en un único sistema, es más fácil gestionar la información.
- **Mejor planificación:** Con la información que ofrece el ERP, permite hacer estudios de planificación.

3.3.1 Desventajas:

- **Coste inicial y recurrentes:** Implantar un ERP supone un coste inicial elevado en equipos, revisión de procesos etc. Además, una vez implantado se tiene que abonar licencias y cuotas de soporte o nube en función del modelo seguido.
- **Desarrollo descoordinado:** Debido a que el ERP permite que se desarrollen las distintas áreas de la empresa, puede surgir de que o un área se desarrolle más que otra, creando un desequilibrio interno, o bien que se desarrollen todas a la vez y haga que los procesos de mejora sean más lentos y menos flexibles.

- **Incompatibilidad:** Puede ser que otros agentes, proveedores, o clientes, no trabajen con herramientas ERP y puede reducir la capacidad del ERP de nuestra empresa.
- **Migración de Datos:** Es necesario un buen plan de migración de datos para evitar pérdidas de información.
- **Formación Empleados:** Al generar nuevos procesos en la empresa, es necesario que los empleados reciban formación acerca de cómo usarlo, y evitar la resistencia al cambio.

3.4 Tipos de ERP

En función de las características que tenga nuestra empresa, podremos implantar un tipo de ERP que mejor se amolde. Hoy en día, existen en el mercado distintas opciones de ERP para nuestra empresa, las clasificaremos según una serie de características.

3.4.1 Diseño:

- **A medida:** Las empresas de mayor tamaño, con necesidades más complejas y mayor número de departamentos, optarán por un ERP a medida adaptado a sus características.

Cuenta con funcionalidades específicas de la compañía, aunque el coste puede ser elevado

- **Predefinido:** Si nuestra empresa está dentro de las PYMES, podemos contar con un ERP predefinido que ofrece soluciones estándar.

Este tipo de ERP también cuenta con una estructura modular y de un precio asequible para nuestra empresa, aunque será necesario encontrar un proveedor que nos garantice una adecuada integración.

3.4.2 Alojamiento:

- **Local:** Nuestra empresa puede contar con un servidor propio donde poder alojar nuestro ERP.

Esta opción nos aporta mayor seguridad y confidencialidad de nuestros datos, pero a su vez somos responsables de su integridad.

- **En la nube o SaaS:** Nuestro ERP se alojará en un servidor externo que pertenezca a los propios proveedores del software.

Nos permite mayor agilidad para acceder al programa desde cualquier lugar con acceso a internet, sin embargo, con esta opción no tenemos un control directo sobre nuestros datos.

3.4.3 Solución:

- **Horizontal:** Este tipo de ERP son generalistas, pues abarcan una serie de prestaciones válidas para todo tipo de empresas, aunque posteriormente se pueda ajustar mediante módulos.

La ventaja es que es útil para todo tipo de empresas, aunque si queremos personalizarlo puede encarecerse o que tengamos que recurrir a otros programas.

- **Vertical:** Estos son sistemas sectoriales, pensados para cubrir las necesidades en función del sector de nuestra empresa.

La ventaja es que, al estar pensado en nuestro sector, ofrece funciones especializadas para mejor integración, sin embargo, son soluciones menos flexibles y más costosas.

3.4.4 Desarrollo:

- **Código Abierto:** Este tipo de ERP permite que su uso sea gratuito, basado en la colaboración abierta donde el usuario se descarga el código fuente directamente.

Las ventajas de este producto son:

- *Coste de adquisición "0" o muy bajo.*
- *Permite añadir funcionalidades durante el tiempo.*

Pero encontramos una serie de desventajas como:

- *Garantías y coberturas de terceros.*
- *Producto menos adaptado a nuestra empresa.*
- **Código privado o de propietario:** Este producto es el modelo tradicional que exige un pago para su uso. Suele estar enfocado a grandes empresas con necesidades específicas.

Las ventajas de este producto son:

- *Ofrece mayor garantía en su desarrollo, funcionamiento, modularidad y actualizaciones.*
- *Soporte frente fallos en el sistema.*
- *Implantación realizada por personal capacitado.*

Sin embargo, tiene una serie de desventajas como:

- *Coste elevado de compra, licencia y servicios.*
- *Dependencia de uso por parte del proveedor.*

| | | |
|--------------------|--|--|
| Diseño | ERP a medida <ul style="list-style-type: none"> • Adaptado a la empresa • Coste más elevado | ERP predefinido <ul style="list-style-type: none"> • La empresa se adapta a él • Coste más asequible |
| Alojamiento | ERP local <ul style="list-style-type: none"> • Mayor seguridad • <i>On premise</i>, acceso limitado | ERP en la nube <ul style="list-style-type: none"> • Datos en servidores ajenos • Agilidad y accesibilidad |
| Solución: | ERP horizontal <ul style="list-style-type: none"> • Para todas las empresas • Menos funciones específicas | ERP vertical <ul style="list-style-type: none"> • Para empresas de un sector • Más rígidas y costosas |
| Desarrollo | ERP código abierto <ul style="list-style-type: none"> • Gratuitos • Complicada personalización | ERP código privado <ul style="list-style-type: none"> • Pago por licencia o cuotas • Garantía en uso y desarrollo |

Ilustración 4: Matriz ERP

Fuente: SoftDoit

4. La Empresa

En este apartado explicaré la situación previa al ERP en la que se encuentra la empresa, detallando sus problemas y necesidades, y como se gestiona el negocio actualmente.

4.1 Prólogo

Antes de empezar, la empresa que me ha ayudado en este trabajo quiere que sus datos en este trabajo queden anonimizados, por ello, toda información delicada de la empresa será reservada únicamente para este trabajo académico.

Para poder referirnos a la empresa usaremos el nombre de DiseñAlmería.

4.2 La Empresa

La empresa en cuestión es una PYME llamada DiseñAlmería, dedicada a proyectos de Diseño Técnico, Textil e Interiorismo.

La empresa cuenta con una tienda física en el centro de Almería donde atender a los clientes, además de estar adscritos a dar coberturas de los seguros de hogar y clientes particulares por teléfono.

En dicha tienda cuentan con: material de exposición, una zona para atender a los clientes que quieran un diseño personalizado, y también sirve como punto de recogida por si un cliente quisiera material de la empresa e instalarlo por sí mismo.

Con la llegada de la Covid-19, la tienda física cerró, de modo que tuvieron que renovar su modelo de negocio.

La empresa cuenta con 4 trabajadores: Primero de todo tenemos a la jefa Principal de la Empresa, una mujer polifacética encargada de la contabilidad de la empresa, atención a los clientes, instalación, realización de diseños y gerente de la empresa.

Después tenemos 1 Instalador técnico, que se encarga de realizar las instalaciones de los trabajos que se encarguen.

Por otra parte, contamos con 2 ayudantes técnicos que sirven de apoyo al instalador principal si lo necesite y también encargados de transportar el material durante una instalación si fuera necesario.

4.3 Situación Previa

Para poder garantizar un cambio óptimo, es necesario identificar correctamente de dónde venimos para saber a dónde queremos ir.

Para ello, se realizó una reunión con la gerente de la empresa y 1 de los técnicos de la empresa, con el objetivo de identificar las necesidades existentes y los procesos de negocio de la empresa previos al ERP.

Después de dicha reunión, se consigue visualizar el modo de trabajo de la empresa y puntos de mejora que podrían ser resueltos con un ERP.

Gracias a esta reunión, se planifica una estrategia de implantación, algo que como se ha explicado anteriormente, es clave para una implantación satisfactoria.

4.4 Procesos de la empresa

En este apartado explicaremos los distintos procesos existentes de la empresa, tanto internos como externos, que podemos mejorar o gestionar de una forma más eficiente gracias a la implantación de un ERP.

4.4.1. Gestión de Clientes

Según la jefa principal de la empresa, esta actividad es la más importante pues “Sin clientes, no recibo ingresos”, según palabras textuales de ella.

Para captar clientes potenciales lo hacen mediante el método de “Puerta Fría”, como se hace desde hace años, yendo a las instalaciones de clientes o contactando a través de Redes sociales. Si bien estos clientes pueden suponer una buena parte de la facturación de la empresa, el método no es siempre el más efectivo, pues en algunas ocasiones estos clientes no revisan sus redes sociales o no retienen toda la información de interés de la empresa.

También tenemos otros 2 tipos de clientes que tiene la empresa, que se diferencian entre **Particulares** y **Asegurados**.

Los **particulares** son clientes que contactan por referidos o por antigua relación y los clientes **asegurados** que vienen por parte de seguros de hogar que ofrecen coberturas a sus clientes y esta empresa los soporta.

En función del tipo de cliente, se gestiona de una forma u otra, priorizándose unos y otros en función de cliente o proyecto.

Además de la captación, les interesa poder recopilar información de los clientes, potenciales o no, para poder fidelizarlos o recopilar información para futuras estrategias de ventas.

4.4.2 Contabilidad y Facturación

La mayor parte de la gestión de la contabilidad recae en la figura del asesor de la empresa. Se tiene que reflejar la información tanto de compras, gastos, ventas, balances y etcétera.

Se realiza a través de facturas en papel, donde se anotan los datos, de cliente o proveedor, y se envía al asesor y a su programa Contasol mediante una foto o se anota en una agenda personal.

Dado que en esta empresa la contabilidad se guarda de forma externa, no siempre se tiene la información completa, ya que no puede consultar ContaSol en el móvil cómodamente y para ello tiene que usar su ordenador local expresamente.

La Directiva de la empresa indicó explícitamente que está descontenta con su programa de gestión pues si realiza alguna gestión desde el móvil no está correctamente sincronizada.

4.4.3 Gestión de Compras

Las compras se gestionan mediante facturas, y éstas se guardan en físico y se mandan al asesor a través de una foto.

El problema es que, si la factura en papel o la agenda que usan para anotarla pueden perderse perfectamente, y las que se envían por correo electrónico no tienen una clasificación adecuada para poder tratarlas.

Dado que el modelo de negocio de la empresa no requiere de un gran volumen de inventario, ellos compran y venden en el mismo día o lo realizan para disponer del inventario durante un breve periodo, por lo que se generan facturas diariamente que tienen que ser correctamente tratadas, asegurando su integridad.

4.4.4 Gestión de Ventas

Las ventas de la empresa se realizan de igual forma que las compras, a través de facturas de papel y presupuestos para los clientes.

Estos presupuestos no tienen un formato dinámico, y para poder realizar un presupuesto lo más cierto posible, es necesario llamar previamente al proveedor para determinar el precio de los materiales, pues no se tienen registrados ni actualizados los precios de los materiales.

4.5 Resumen Necesidades

Dada la edad de los trabajadores de la empresa y el modelo de negocio, podemos decir que la digitalización de la empresa es mínima, y son conscientes de que si dieran el salto digital podrían potenciar estrategias de venta que tienen previstas y poder recapitular toda la información de la empresa para sus propios análisis internos, ya que actualmente no se recopila y lo que se tiene puede perderse fácilmente.

Tras haber estudiado las necesidades de la empresa, concluimos que los puntos principales de cambio son

- ***Gestión de la información:*** Dado que existe alta probabilidad de pérdida de información e interés por parte de la empresa de aprovechar dicha información, se considera contar con un programa para recopilar y almacenar la información para tratarla posteriormente.
- ***Eficiencia de gestión:*** Los procesos de negocio de la empresa pueden mejorarse gracias a un ERP, a la par que permite tener alto detalle de los distintos procesos.
- ***Digitalización:*** La empresa reconoce que el mundo digital es presente y futuro, y quieren explotar la digitalización para poder llegar mejor a clientes.

5. Selección del ERP y Planificación del Proyecto

5.1 Selección del ERP

Tras haber introducido anteriormente los tipos de ERP existentes en el mercado (Software Libre, propietario y SaaS) y haber detallado el contexto de la empresa con la que vamos a trabajar, es momento de decidir qué tipo de ERP es el más apropiado.

Merece recordar que la empresa consta de 4 trabajadores y que por ahora no cuenta con grandes instalaciones físicas, además que dada la facturación y actividad que realiza la empresa, el presupuesto destinado para la implantación de un ERP está limitado.

Por tanto, la opción más adecuada para la empresa es un ERP pueda resolver las necesidades principales de la empresa, que no son demasiado complejas, y que esté enfocado para las PYMES permitiendo mayor adaptabilidad a la empresa.

Además, primeramente, se precisaría de un ERP de tipo local que permita reducir el coste y evitar pagar un suplemento por tenerlo alojado en un servidor externo.

Teniendo en cuenta todo lo mencionado anteriormente y habiendo analizado las ventajas y desventajas que ofrecen cada uno de ellos, la elección más adecuada es la de un ERP Libre.

5.2 Comparativa y Selección.

Una vez seleccionado el ERP, haremos una comparativa de los ERP de tipo Libre con mayor presencia en el mercado. Tras exponer las características principales de los mismos, seleccionaremos el que más se adecúe al tipo de negocio que tenemos.

5.2.1 Apache OFBiz

En esta ocasión, Apache OFBiz más que un Software ERP es un software de automatización que compromete un conjunto de aplicaciones integradas y que automatiza los procesos de negocio de una empresa.

Es un framework basado en Java y que su estructura permite a los desarrolladores ampliar y mejorar su framework de forma personalizada.

La estructura de Apache OFBiz está construida sobre un framework común, que está dividida en distintas capas.

Capa de Presentación: Se refiere puntualmente a las páginas webs que se muestran en pantalla ante el cliente. Cada página está formada a partir de componentes que pueden ser Java Server Pages, consultar, widget y menús. El conjunto de todo lo anterior especifica la definición de pantalla.

Capa de Negocio: Se refiere a la capa empresarial que está orientada a la visión de usuario. Esta capa está pensada para que funcione por métodos JAVA, flujos de trabajo o servicios simples.

Capa de Datos: Esta capa es la responsable de acceso a los datos. No se accede a los datos de forma orientada a objetos, sino que de forma relacional.



Ilustración 5: Apache OFBiz

5.2.2 Adempiere

Adempiere es un sistema ERP de código abierto orientado a empresas de tamaño pequeño y mediano. Está soportado por desarrolladores de Adempiere y se constituye como un ERP que, en este caso, su arquitectura está basada en diccionario de datos y flujos de trabajo, lo que permite que la personalización del sistema puede ser realizado en gran parte sin la generación de nuevo código

Entre sus características principales están:

- ERP modular: Cuenta con variedad de módulos como control de inventario, gestión financiera, RRHH, punto de venta, SCM, CRM y tienda online.
- Su sistema está basado en Oracle y PostgreSQL.
- Permite integración con otros sistemas software de la empresa.

- Puesto que Adempiere es de código abierto, es un sistema fácil de adaptar pudiendo editar el código fuente según las necesidades de la empresa.

- Adempiere ofrece documentación gratuita para desarrolladores.



Ilustración 6: Adempiere

5.2.3 Odoo

Odoo es un software ERP integrado bajo licencia LGPLv3. A parte de ser un ERP, incorpora herramientas como CRM, comercio electrónico, web y módulos de personalización. Este software ha sido desarrollado por Odoo S.A., quien declara su producto como una alternativa de código abierto a SAP ERP o Mycrossoft Dynamics.

En la actualidad, Odoo cuenta con más de 46 módulos diferentes como la gestión de ventas, CRM, gestión de proyectos, gestión de almacén, punto de venta, inventario, flujos de trabajo, entre otros.

Dado que este software está dividido en módulos, la variedad de que ofrece permite alternativas de personalización para que las empresas puedan seleccionar los que más se adapten a su modelo de negocio.

Su bajo coste de adquisición y alta capacidad de personalización la convierte en una de las opciones más atractivas para el mundo de las PYME.

Odoo ofrece en su página web una gran cantidad de información y documentación de su software tanto en forma de manuales de usuario como vídeos explicativos, además de que existe una gran comunidad colaborativa para el desarrollo de este software.

Dado que este ERP es de código abierto, se puede acceder al código fuente para adaptarlo a la organización sin necesidad de contactar con el distribuidor, pudiendo implementar las nuevas funcionalidades que se deseen.



Ilustración 7: Odoo

5.3 Análisis y Selección.

Para poder hacer una correcta selección del software ERP, se ha realizado una reunión con la gerente de la empresa para analizar las necesidades principales de la empresa y, junto con las características principales de los distintos sistemas ERP que se han presentado, hacer una matriz de selección.

La matriz empleada para la selección del software ERP es la siguiente:

| Necesidades | Apache OfBiz | Adempiere | Odoo |
|---|---------------------|------------------|-------------|
| Gestión de Procesos | Sí | Sí | Sí |
| Compras, Inventario, Contabilidad y Ventas | Sí | Sí | Sí |
| CRM | Sí | Sí | Sí |
| Multiplataforma | No | No | Sí |
| Modularidad | Sí | Sí | Sí |
| Uso en dispositivos móviles | No | No | Sí |
| Licencia Personalizada | No | No | Sí |
| Gestión de Proyectos | No | No | Sí |

Después de haber presentado a la gerente de la empresa la matriz de selección y haber discutido sobre las ventajas y desventajas que ofrecen cada alternativa, se ha decidido por implantar el Software ERP Odoo ya que es la herramienta que mayor necesidad cubre y que mayor opción ofrece a la empresa. Por ejemplo, Odoo cuenta con una herramienta de Gestión de Proyectos que para el tipo de empresa que se trata es una necesidad prioritaria para su correcto desarrollo.

5.4 Planificación y Estrategia.

Tras haber decidido el software a implantar en la empresa, es preciso diseñar una planificación donde reflejar las distintas fases del proyecto, planificando el tiempo estimado para cada una de ellas con el objetivo de realizar un correcto desarrollo.

En este apartado se expone brevemente el plan de acción llevado a cabo para este proyecto con la entidad:

- La planificación para este proyecto es de camino crítico ya que existe dependencia entre los bloques de trabajo, por lo que es importante que ambas partes cumplan en la mayor parte posible las fechas comprometidas en este documento.
- Este plan de acción es una propuesta y será lo primero a reformular junto con los responsables de la entidad para asegurarnos que damos prioridad a la funcionalidad más importante de la entidad.
- Debido a que se trata de un proyecto vivo, la planificación que se propone es orientativa. Después podrán ir variando fechas dependiendo del avance o las necesidades de los usuarios.
- El proyecto tendrá una duración aproximada de 6 meses a partir de la aceptación de la oferta por parte de la entidad para el arranque del proyecto, asumiendo la responsabilidad por ambas partes de establecer reuniones periódicas, de forma presencial o vía teams, para el desarrollo del proyecto de implantación.
- Una vez arrancado se dará soporte presencial al arranque y formación del Software en las instalaciones de la entidad y pasado ese mes de soporte presencial se dará soporte vía telemática.

Tras haber expuesto el plan de acción, se presenta el Diagrama de Gantt de la planificación prevista para este proyecto:

Gantt Chart

Planificación de la estrategia de implantación del Software Odoo en el periodo 2022-2023

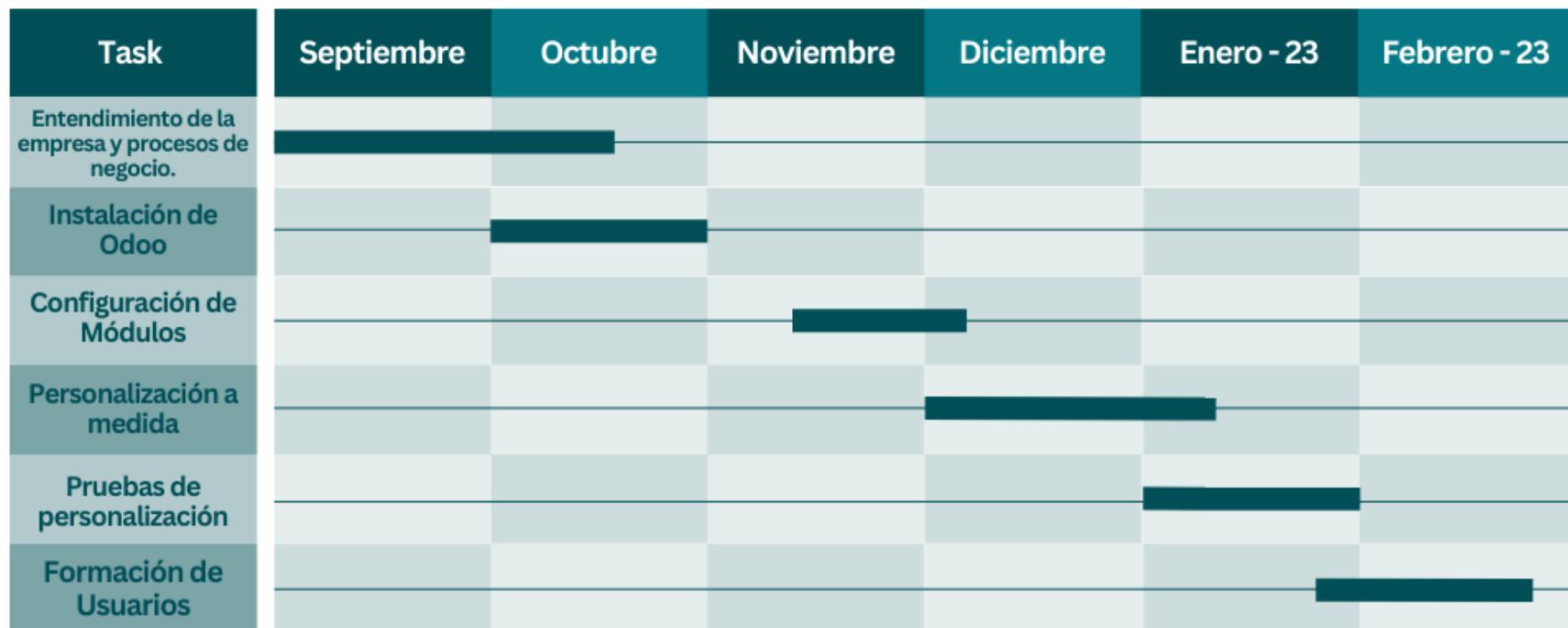


Ilustración 8: Diagrama de Gantt

5.5 Coste del Proyecto.

Respecto al apartado económico, se expondrá los principales aspectos que tienen impacto en el presupuesto para poner en marcha el proyecto para la entidad: Licencia del producto; tiempo de consultoría; tiempo de desarrollo e implementación, formación de los usuarios finales y mantenimiento.

- **Licencia:** Es el coste de licencia de uso del Software, además, en algunos, incluye servicios de mantenimiento. En este aspecto el coste es 0 ya que Odoo no requiere de licencias.
- **Consultoría:** En este apartado entra el coste asociado a la labor del análisis previo y entendimiento de la entidad, la instalación del software y otros aspectos de desarrollo durante el proyecto.
- **Desarrollo e Implementación:** En este apartado se tiene en cuenta los aspectos de la personalización, la realización de las pruebas pertinentes y su puesta en marcha.
- **Formación de Usuarios:** En este apartado se centra el coste de formación a los usuarios finales, siendo un aspecto inherente pues garantiza el éxito de implantación y satisfacción del cliente.
- **Mantenimiento:** En este apartado se refleja el coste del mantenimiento post-impantación previsto por futuras necesidades de cambio o actualizaciones.
- **Hardware:** Dado que ya cuentan con ordenadores para instalar el servidor y dispositivos móviles, este aspecto no refleja ningún impacto.

| Concepto | Porcentaje del Proyecto | Coste |
|-----------------------------|-------------------------|-------|
| Licencia | 0 | 0 |
| Consultoría | 40 | 8€/h |
| Desarrollo e Implementación | 40 | 10€/h |
| Formación Usuarios | 10 | 8€/h |
| Mantenimiento | 10 | 10€/h |

6. Instalación de Odoo

6.1 Instalación:

Para la instalación del software Odoo ERP no va a ser necesario tener que realizar nuevas adquisiciones tecnológicas. Dado que Odoo es una herramienta multiplataforma, se instalará en el ordenador principal de la empresa y en caso de querer acceder desde otra plataforma (móvil, Tablet...) se podrá acceder siempre y cuando éste ordenador esté conectado a la red.

Dado que Odoo está pensado para ser usado por pequeñas y medianas empresas, que no necesariamente tienen que disponer de equipos tecnológicos de grandes características, no pone restricciones para su instalación en cuanto a software y hardware se refiere. Sin embargo, sí que recomienda unos requisitos mínimos para garantizar el correcto funcionamiento del software.

En este caso en particular, el software va a ser instalado en un equipo de Windows 10 con 8 GB de RAM y una memoria de 1TB de tipo HDD.

Expuesto todo lo anterior, el primer lugar es acceder a la página de Odoo y proceder a su descarga: (https://www.odoo.com/es_ES/page/download)

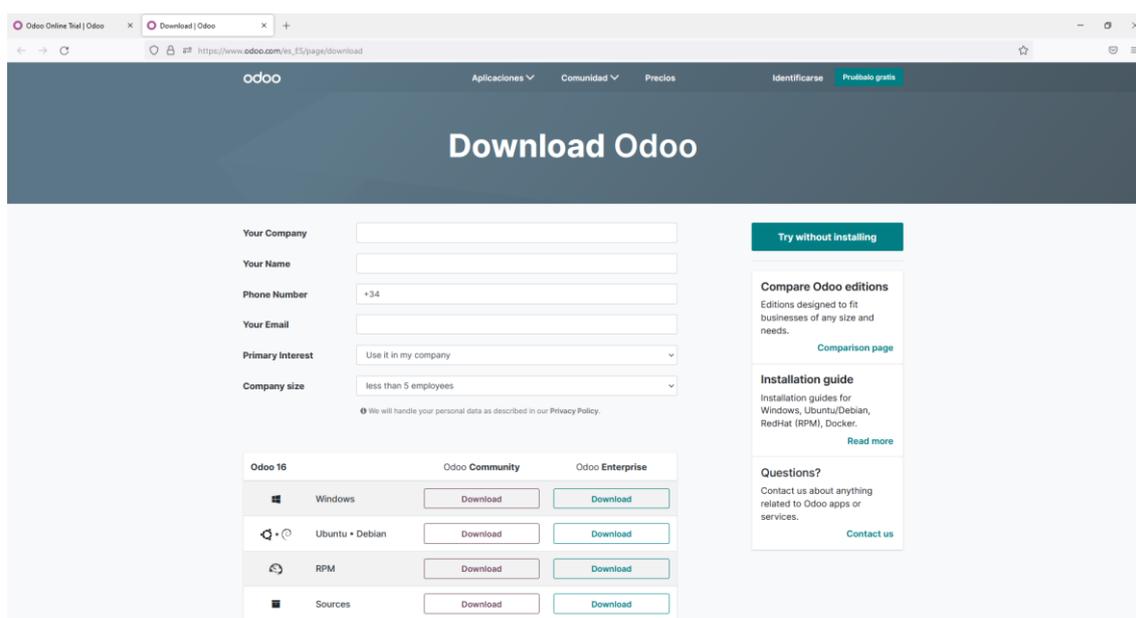


Ilustración 9: Descarga de Odoo.

Para poder proceder a la descarga será necesario detallar la información de la empresa. En este caso se va a instalar la versión Odoo 14 pues es una versión muy estable y que dispone de muchos recursos en la comunidad de Odoo en caso de necesitar soporte.

| Odoo 14 | Odoo Community | Odoo Enterprise |
|---|--------------------------|--------------------------|
|  Windows | Download | Download |
|  Ubuntu • Debian | Download | Download |
|  RPM | Download | Download |
|  Sources | Download | Download |

Ilustración 10: Versión de Odoo

Tras descargar y ejecutar el instalador nos pide seleccionar el idioma.

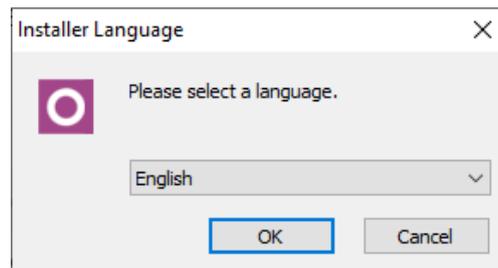


Ilustración 11: Idioma de instalación.

En la siguiente ventana nos muestra el tipo de instalación que queremos realizar, ofreciéndonos 3 tipos:

-All in One: Instalación de servidor Odoo y Base de Datos PostgreSQL, es la opción más sencilla.

-Server Only: Esta opción nos instalaría únicamente el servidor de Odoo, muy útil en caso de tener un gestor de base de datos o que las bases de datos estén alojadas en otro servidor.

-Custom(Personalizado): Esta opción permite instalar los componentes por separado, dando más opciones al usuario.

Para esta instalación se ha seleccionado la opción “All in One” ya que esta opción nos proporciona ambas herramientas, Servidor y Base de Datos.

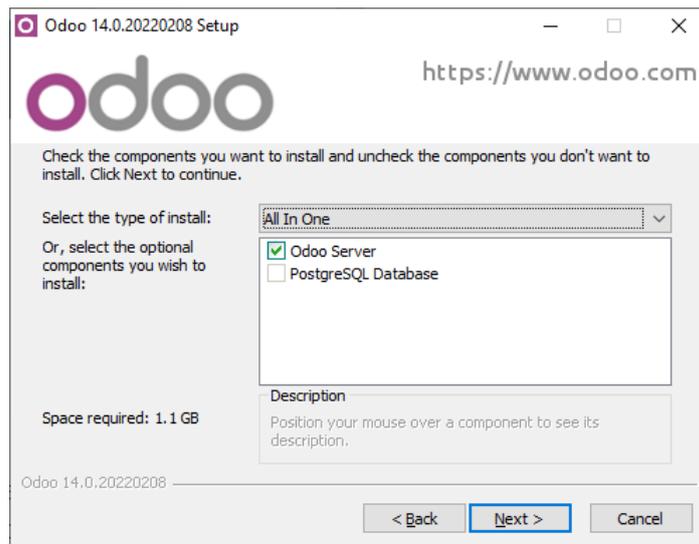


Ilustración 12: Tipo de Instalación

Después de haber seleccionado el tipo de instalación tenemos que configurar la conexión con PostgreSQL.

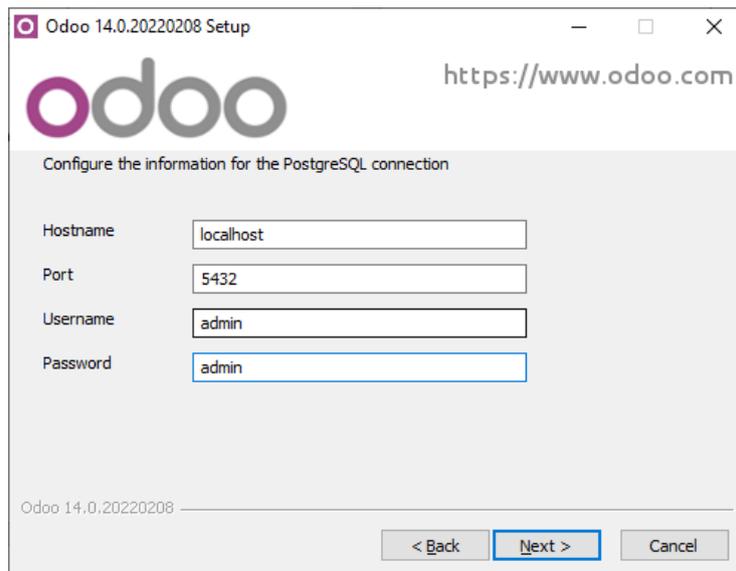


Ilustración 13: Conexión PostgreSQL

En esta ventana hay que indicar:

- Hostname:** Nombre del servidor donde se va a instalar el software. Dado que se va a instalar en local, lo pondremos como “localhost”
- Port:** Puerto de acceso a la base de datos.
- Username:** Nombre del usuario que accede al sistema.
- Password:** Clave de acceso del usuario.

Por último, se elige el directorio en donde se instalará todo el paquete. En este caso se respetará la ruta que se indica por defecto.

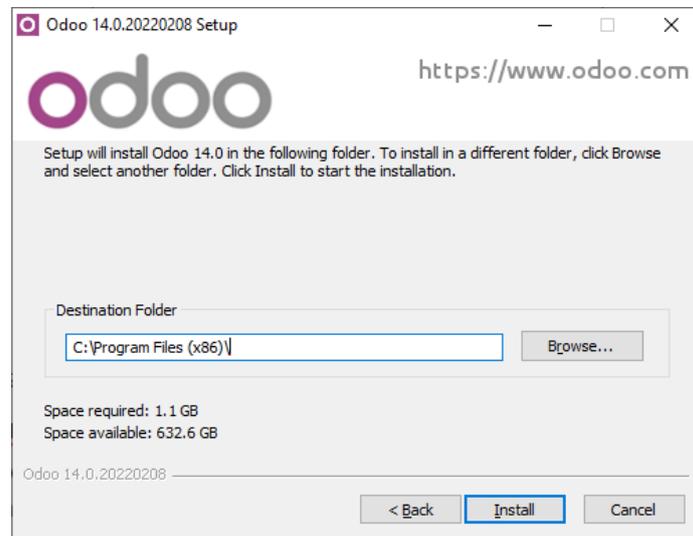


Ilustración 14: Ruta directorio.

Finalmente habremos conseguido instalar nuestro Software ERP Odoo en nuestro equipo. Ahora, para el primer acceso al software es necesario entrar desde nuestro navegador web introduciendo la dirección IP y el puerto local de equipo donde se encuentra instalado.

Para este equipo la dirección es la <http://localhost:8069>. En caso de que queramos acceder desde otro dispositivo que estuviera en la misma red habría que poner la dirección <http://IP Ordenador Odoo:8069>

Introduciremos la dirección en nuestro ordenador local y accederemos al servidor Odoo. En dicho servidor, generaremos la base de datos de nuestra empresa con la información relativa a la misma.

Create Database ✕

Master Password 👁

Database Name

Email

Password 👁

Phone number

Language ▾

Country ▾

Demo data

To enhance your experience, some data may be sent to Odoo online services.
See our [Privacy Policy](#).

Continue

Ilustración 15: Creación BBDD

Una vez completado, nos aparecerá la ventana de acceso a la base de datos.

Your logo

Base de datos

DA Select

Correo electrónico

Contraseña

Iniciar sesión

[Gestionar Bases de datos](#) | [Respaldo por Odoo](#)

Ilustración 16: Acceso BBDD

Una vez accedemos con las credenciales definidas nos aparece la siguiente ventana.

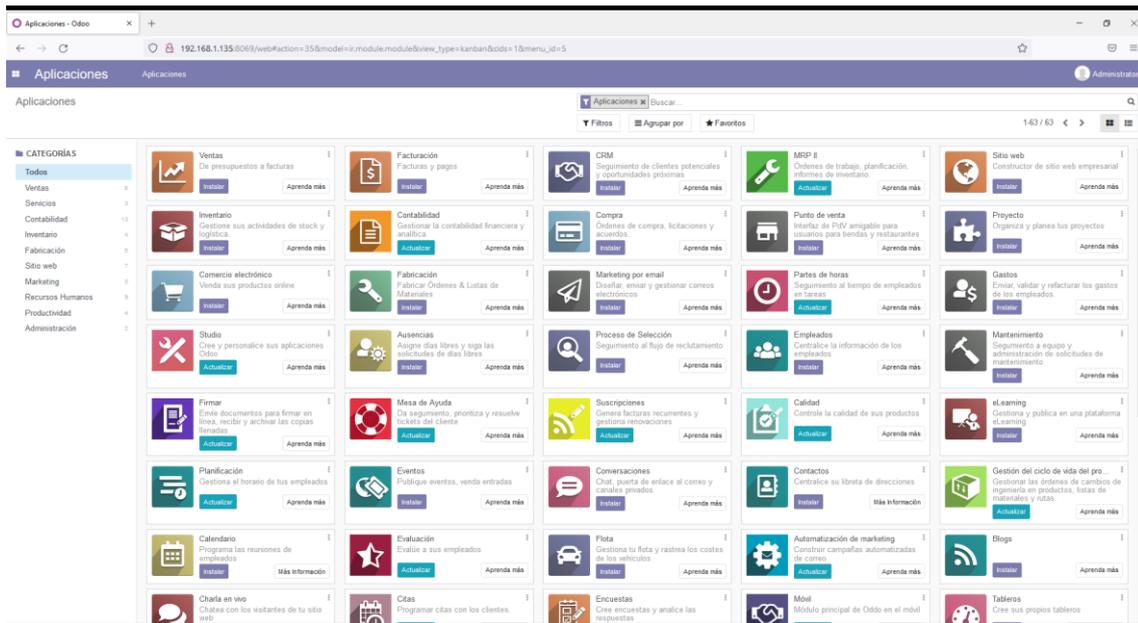


Ilustración 17: Listado de Módulos

6.2 Configuración Inicial:

Una vez hemos accedido a la base de datos, para hacer la configuración inicial tendremos que definir la información de la empresa para que los módulos que incorporemos estén vinculados con la información de la misma, para ello iremos a la ventana de “Ajustes”.

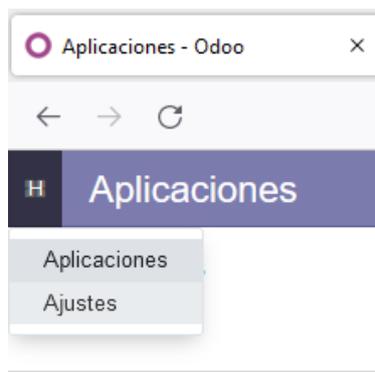


Ilustración 18: Ajustes Odoo

Una vez seleccionamos “Ajustes” nos iremos a la ventana de “Usuarios y Compañía”.



Ilustración 19: Usuarios y Compañías.

En este apartado seleccionaremos “Compañías”.



Ilustración 20: My Company

Dado que inicialmente la base de datos está vacía, crearemos nuestra compañía y rellenaremos con la información de esta.

Ilustración 21: Creación Empresa

6.3 Instalación de Módulos:

Tras haber completado la información de la empresa, se van a instalar los módulos que necesita la empresa para su negocio. Los módulos que instalar son: Gestión de Ventas, Gestión de Compras, Contabilidad, CRM e Inventario.

Para empezar a instalar y configurar un módulo, en la misma ventana del listado de módulos que se ha mostrado anteriormente, se selecciona “Instalar” del módulo correspondiente.

6.3.1 – Instalación Módulo de Ventas:

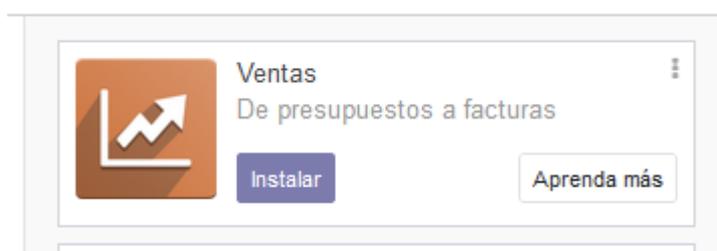


Ilustración 22: Módulo de Ventas

Una vez le hemos dado a instalar, nos aparecerá en el desplegable izquierdo superior de la ventana.



Ilustración 23: Despliegue módulo ventas

Entonces entrando en el apartado ventas podremos entrar en el módulo.

Ventas Pedidos A facturar Productos Informes Configuración

Presupuestos

Crear

Información de compañía

Fije los datos de su compañía para los encabezados y pies de página de sus documentos.

¡Empecemos!

Diseño de Presupuesto

Personaliza la vista de tus presupuestos.

Personalizar

| <input type="checkbox"/> | Número | Fecha de Creación | Cliente | Comercial |
|--------------------------|---------|-------------------|---------------|----------------|
| <input type="checkbox"/> | REF0001 | 08/12/2022 | Thomas Passot | John Miller |
| <input type="checkbox"/> | REF0002 | 06/01/2023 | John Miller | Henry Campbell |

Ilustración 24: Módulo de ventas.

6.3.2 – Instalación Módulo de Compras:

Compra

Órdenes de compra, licitaciones y acuerdos.

Instalar

Aprenda más

Ilustración 25: Módulo de compras.

Una vez le hemos dado a instalar, nos aparecerá en el desplegable izquierdo superior de la ventana.

Conversaciones

Ventas

Compra

Facturación / Contabilidad

Aplicaciones

Ajustes

Ilustración 26: Despliegue módulo compras.

Entonces entrando en el apartado compras podremos entrar en el módulo.

| Compra | | | | | |
|--|----------------|-------------------------|------------------------|---------------------|---|
| Pedidos Productos Informes Configuración | | | | | |
| Solicitudes de presupuesto | | | | | Buscar... |
| Crear | | | | | Filtros Agrupar por |
| Todas las SdP | | 0 | 0 | 0 | Valor medio del pedido (€) |
| Mis SdP | | Por Enviar | En espera | Retrasado | Plazo de compra |
| Mis SdP | | 0 | 0 | 0 | |
| Referencia | Proveedor | Representante de Compra | Fecha límite de pedido | Actividad siguiente | |
| REF0001 | Wendi Baltz | Henry Campbell | | Integer vitae | |
| REF0002 | Thomas Passot | Thomas Passot | hace 14 días | Integer vitae | |
| REF0003 | Carrie Helle | Thomas Passot | | Laoreet id | |
| REF0004 | John Miller | Henry Campbell | | Laoreet id | |
| REF0005 | Henry Campbell | John Miller | | | |
| REF0006 | Carrie Helle | Henry Campbell | | | |
| REF0007 | Carrie Helle | Henry Campbell | | | |
| REF0008 | Thomas Passot | Thomas Passot | | | |

Ilustración 27: Módulo compras.

6.3.3 – Instalación Módulo de CRM:



Ilustración 28: Módulo de CRM

Una vez le hemos dado a instalar, nos aparecerá en el desplegable izquierdo superior de la ventana.



Ilustración 29: Despliegue módulo CRM.

Entonces entrando en el apartado CRM podremos entrar en el módulo.

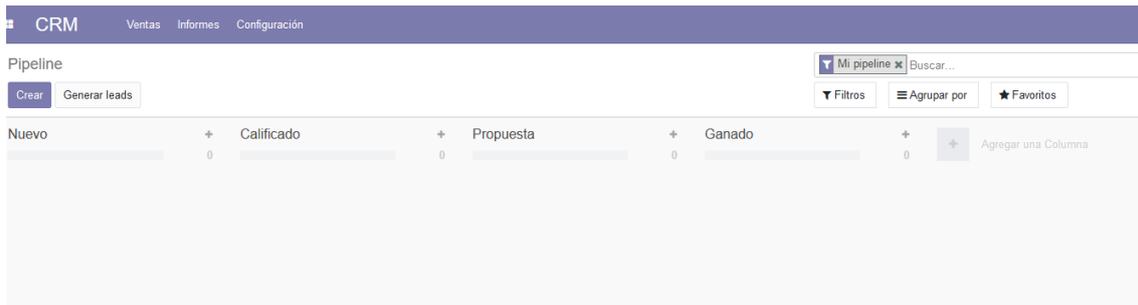


Ilustración 30: Módulo CRM.

6.3.4 – Instalación Módulo de Inventario:



Ilustración 31: Módulo de Inventario.

Una vez le hemos dado a instalar, nos aparecerá en el desplegable izquierdo superior de la ventana.

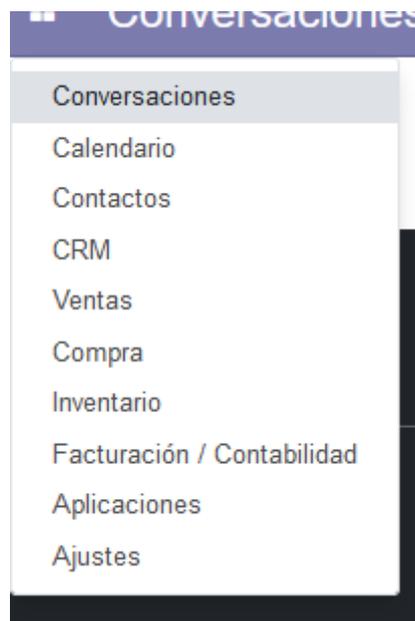


Ilustración 32: Despliegue módulo inventario.

Entonces entrando en el apartado CRM podremos entrar en el módulo.

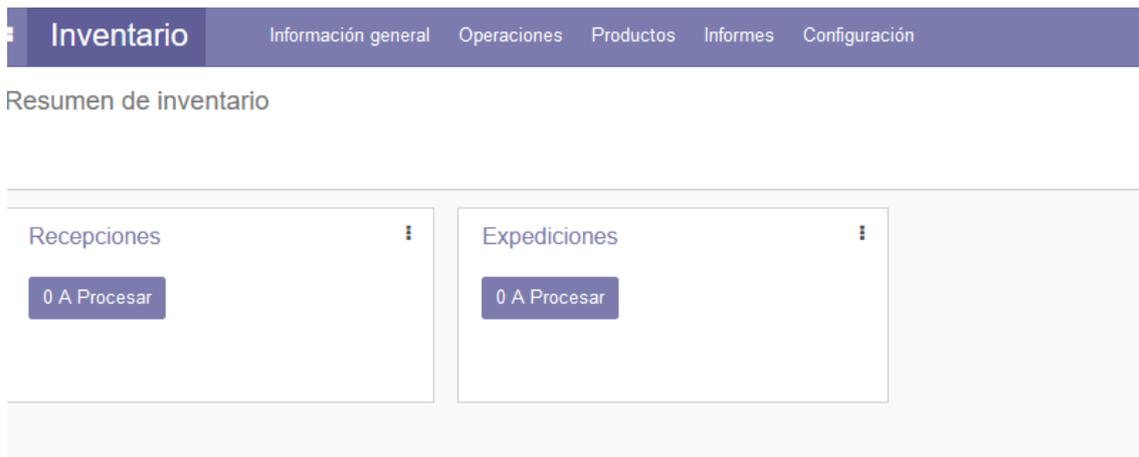


Ilustración 33: Módulo de inventario.

7. Personalización Odoo

7.1 Facturación:

Para empezar a usar la aplicación, la empresa quiere personalizar el modelo de factura para que se ajuste a su imagen empresarial.

En la siguiente captura, podemos ver el modelo de factura que tiene por defecto Odoo

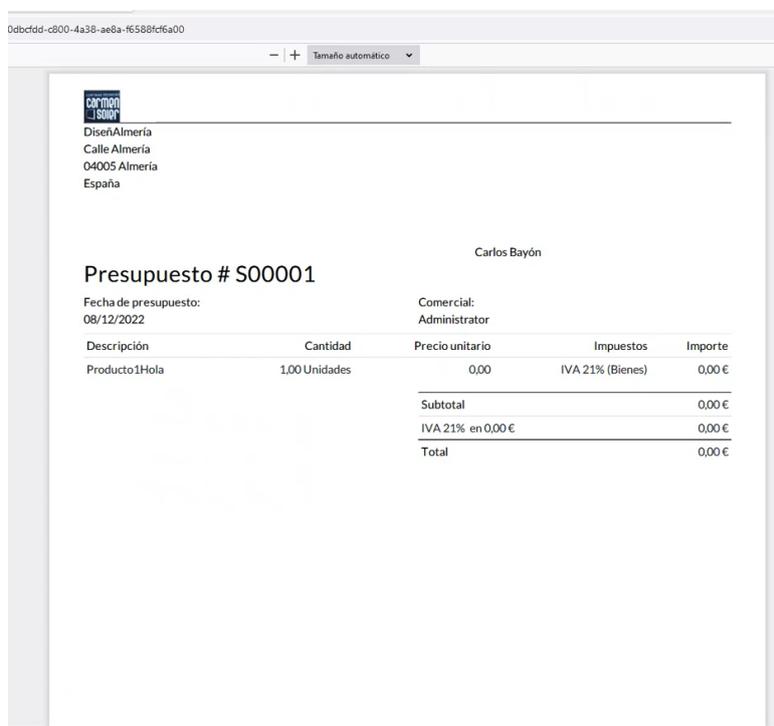


Ilustración 34: Factura Odoo

Gracias a que Odoo es un software de código abierto, nos permite editar el modelo de factura según las necesidades que nuestra empresa demanda.

Antes de empezar, tenemos que entender cómo está compuesta la factura a nivel de SW.

Para la personalización, vamos a tener que trabajar con **Vistas** y **Modelo de Datos**, y lo haremos además en lenguaje Python, que así es como está configurado Odoo.

Los modelos determinan la estructura lógica de una base de datos y cómo se almacenan, organizan y manipulan los datos. En otras palabras, un modelo es una tabla de información que se puede vincular con otras tablas.

Un modelo suele representar un concepto empresarial, como una *orden de venta*, un *contacto* o un *producto*.

Los módulos y las aplicaciones contienen varios elementos, como modelos, vistas, archivos de datos, controladores web y datos web estáticos.

Las vistas son la interfaz que permite mostrar los datos contenidos en un modelo. Un modelo puede tener varias vistas, que simplemente son diferentes formas de mostrar los mismos datos.

Para saber cuál es el fichero que necesitaremos tratar, desde el módulo de Ventas de Odoo, vemos que el fichero de Vista sobre el que vamos a trabajar para la factura que nos muestra es el de "sale_report_template.xml"

| Informes | | | | Buscar | | |
|-------------------------------------|-------------------------------------|------------------------|------------------|---|-------------|-----------|
| Crear | | | | Filtros | Agrupar por | Favoritos |
| <input type="checkbox"/> | Etiqueta del Producto (PDF) | product template | s.actions.report | product_report_productionlabel | | PDF |
| <input type="checkbox"/> | Etiqueta del Producto (PDF) | product product | s.actions.report | product_report_productionlabel | | PDF |
| <input type="checkbox"/> | Etiqueta del producto (PDF) | product product | s.actions.report | stock_label_product_new | | Texto |
| <input type="checkbox"/> | Etiqueta del producto (PDF) | product template | s.actions.report | stock_label_product_template_new | | Texto |
| <input type="checkbox"/> | Extracto | account bank statement | s.actions.report | account_report_statement | | PDF |
| <input type="checkbox"/> | Factura PRO-FORMA | sale order | s.actions.report | sale_report_saleorder_proforma | | PDF |
| <input type="checkbox"/> | Facturas | account move | s.actions.report | account_report_move_with_payments | | PDF |
| <input type="checkbox"/> | Facturas originales | account move | s.actions.report | account_report_original_vendor_bill | | PDF |
| <input type="checkbox"/> | Facturas sin pagar | account move | s.actions.report | account_report_move | | PDF |
| <input type="checkbox"/> | Guía técnica | s.module.module | s.actions.report | base_report_modulereference | | PDF |
| <input type="checkbox"/> | Hoja de cuentas | stock inventory | s.actions.report | stock_report_inventory | | PDF |
| <input type="checkbox"/> | Información general del modelo | s.model | s.actions.report | base_report_modelreference | | PDF |
| <input type="checkbox"/> | Informe de Provisión | product product | s.actions.report | stock_report_product_replenishment | | HTML |
| <input type="checkbox"/> | Informe de Provisión | product template | s.actions.report | stock_report_product_template_replenishment | | HTML |
| <input type="checkbox"/> | Informe de Rutas del Producto | product template | s.actions.report | stock_report_stock_route | | HTML |
| <input type="checkbox"/> | Numero de Lote/Serie (PDF) | stock production lot | s.actions.report | stock_report_sl_label | | PDF |
| <input type="checkbox"/> | Numero de lote / serie (PDF) | stock production lot | s.actions.report | stock_label_sl_template_new | | Texto |
| <input type="checkbox"/> | Operaciones de abastec. | stock picking | s.actions.report | stock_report_picking | | PDF |
| <input type="checkbox"/> | Paquete del producto (PDF) | product packaging | s.actions.report | stock_label_product_packaging_new | | Texto |
| <input type="checkbox"/> | Pedido de compra | purchase order | s.actions.report | purchase_report_purchaseorder | | PDF |
| <input checked="" type="checkbox"/> | Presupuesto / Pedido | sale order | s.actions.report | sale_report_saleorder | | PDF |
| <input type="checkbox"/> | Recibón de pago | account payment | s.actions.report | account_report_payment_receipt | | PDF |
| <input type="checkbox"/> | Resultado de integridad de hash PDF | res company | s.actions.report | account_report_hash_integrity | | PDF |
| <input type="checkbox"/> | Solicitud de presupuesto | purchase order | s.actions.report | purchase_report_purchaseorder | | PDF |
| <input type="checkbox"/> | Tarifa | product product | s.actions.report | product_report_pricecost | | PDF |
| <input type="checkbox"/> | Tipo de operación (PDF) | stock picking type | s.actions.report | stock_report_picking_type_label | | PDF |
| <input type="checkbox"/> | Tipo de operación (PDF) | stock picking type | s.actions.report | stock_label_picking_type_new | | Texto |
| <input type="checkbox"/> | Visa previa del Informe Externo | res company | s.actions.report | web_preview_externalreport | | PDF |
| <input type="checkbox"/> | Visa previa del Informe Interno | res company | s.actions.report | web_preview_internalreport | | PDF |

Ilustración 35: Modelo Presupuesto Odoo

Y para esta vista, vemos que el modelo que emplea es el “sale.order” y la plantilla de la vista por defecto es la “sale.report_saleorder”

Tradiciones Técnico
Acción

Elminar del menú "Tradiciones"
Vistas QWeb

| | | | | |
|------------------|----------------------|----------------------------|---|--|
| Nombre | Presupuesto / Pedido | | | |
| Tipo de informe | PDF | Nombre del modelo | sale.order | |
| Formato de papel | | Nombre de plantilla | sale.report_saleorder | |
| | | Nombre del informe impreso | (object state in ('draft', 'sent') and 'Presupuesto - %s' % (object name)) or 'Pedido - %s' % (object name) | |

Seguridad
Propiedades avanzadas

Nombre del grupo

Ilustración 36: Modelo y Plantilla Odoo

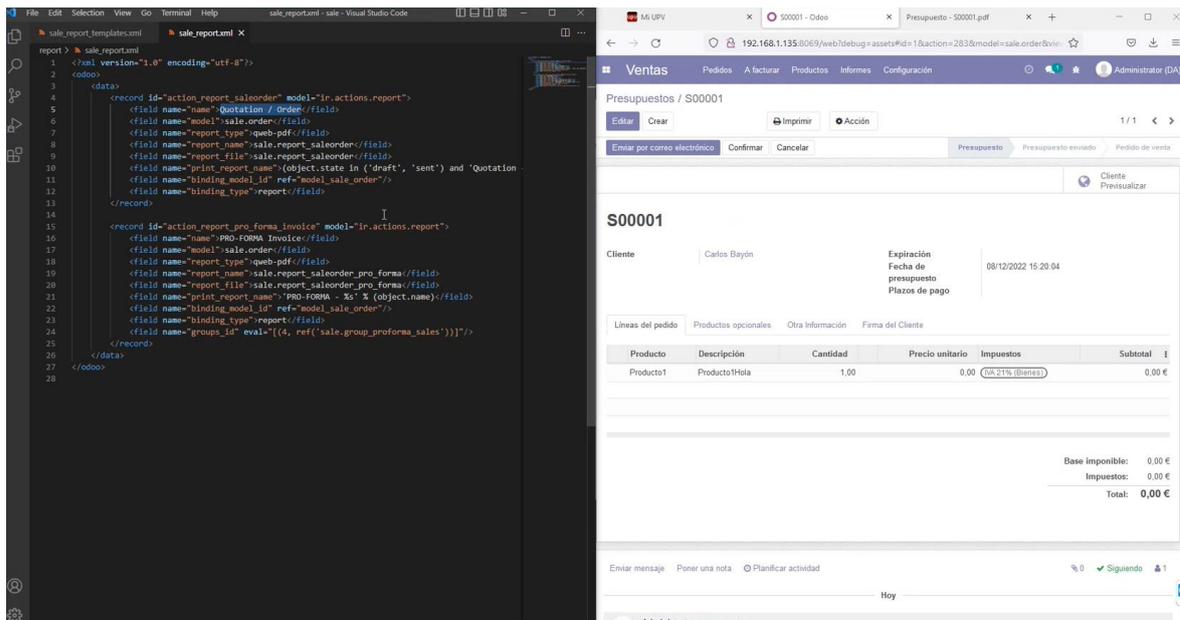


Ilustración 37: Vista Modelo y Plantilla Odoo.

Desde “Vistas QWeb” podemos ver el nombre del modelo y de las plantillas que emplea nuestra vista para poder identificarlo a nivel de código.

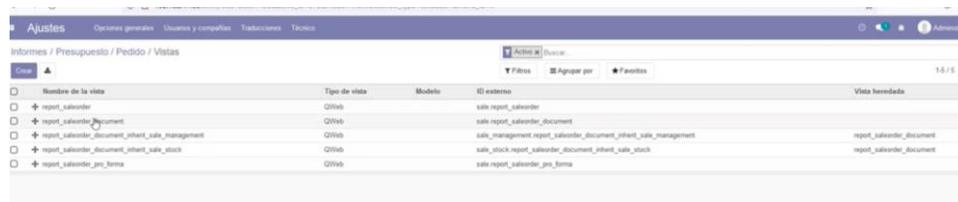


Ilustración 38: Vista QWEB

Una vez identificados, vamos a entender los distintos tipos de acciones e informes que hay dentro del módulo de “sale_order”.

Por ejemplo, en el primer caso, el id = “action_report_saleorder” indica las acciones que hará la vista. Podemos ver que el modelo que va a emplear es el modelo = “sale.order” y el campo el report name = “sale.report_saleorder”. Este reporte lo encontrará en el módulo de ventas “sale.report_template”.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <data>
4 <record id="action_report_saleorder" model="ir.actions.report">
5 <field name="name">Quotation / Order</field>
6 <field name="model">sale.order</field>
7 <field name="report_type">qweb-pdf</field>
8 <field name="report_name">sale.report_saleorder</field>
9 <field name="report_file">sale.report_saleorder</field>
10 <field name="print_report_name">(object.state in ('draft', 'sent') and 'DisenAlmeria - %s' % (object.name)) or 'DisenAlmeria - %s' % (object.name)</field>
11 <field name="binding_model_id" ref="model_sale_order"/>
12 <field name="binding_type">report</field>
13 </record>
14
15 <record id="action_report_pro_forma_invoice" model="ir.actions.report">
16 <field name="name">PRO-FORMA Invoice</field>
17 <field name="model">sale.order</field>
18 <field name="report_type">qweb-pdf</field>
19 <field name="report_name">sale.report_saleorder_pro_forma</field>
20 <field name="report_file">sale.report_saleorder_pro_forma</field>
21 <field name="print_report_name">'PRO-FORMA - %s' % (object.name)</field>
22 <field name="binding_model_id" ref="model_sale_order"/>
23 <field name="binding_type">report</field>
24 <field name="groups_id" eval="[(4, ref('sale.group_proforma_sales'))]">
25 </record>
26
27 <record id="pedido_de_venta_disenalmeria" model="ir.actions.report">
28 <field name="name">Informe del pedido de venta</field>
29 <field name="model">sale.order</field>
30 <field name="report_type">qweb-pdf</field>
31 <field name="report_name">sale.custom_saleorder</field>
32 <field name="report_file">sale.custom_saleorder</field>
33 <field name="print_report_name">'Sale Order - %s' % (object.name)</field>
34 <field name="binding_model_id" ref="model_sale_order" />
35 <field name="binding_type">report</field>
36 <field name="paperformat_id" ref="sale_order_report_003068_paperformat" />
37 </record>
38 </data>
39 </odoo>
40

```

Ilustración 39: Vista Modelo y Plantilla Odoo

Y en dicho módulo encontramos el reporte correspondiente “report.saleorder”

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <template id="report_saleorder_document">...
4 </template>
5
6 <template id="report_saleorder">
7 <t t-call="web.html_container">
8 <t t-foreach="docs" t-as="doc">
9 <t t-call="sale.report_saleorder_document" t-lang="doc.partner_id.lang"/>
10 </t>
11 </t>
12 </template>
13
14 <template id="report_saleorder_pro_forma">
15 <t t-call="web.html_container">
16 <t t-set="is_pro_forma" t-value="True"/>
17 <t t-set="docs" t-value="docs.with_context(proforma=True)">
18 <t t-foreach="docs" t-as="doc">
19 <t t-call="sale.report_saleorder_document" t-lang="doc.partner_id.lang"/>
20 </t>
21 </t>
22 </t>
23 </template>
24 </odoo>
25

```

Ilustración 40: Vista "sale_report_template.xml"

Para este caso concreto, la template tiene una llamada a “sale.report_saleorder_document”, que es justo la template anterior, que es la que contiene toda la información para mostrar la página.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <template id="report_saleorder_document">
4 <t-call web.external_layout>
5 <t-set doc=t.value doc.with_context(lang=doc.partner_id.lang) />
6 <t-set address>
7 <div t-fields="doc.partner_id">
8 <t-options widget="contact", fields=["address", "name"], no_marker=True />
9 <p t-if="doc.partner_id.vat"><t-esc="doc.company_id.country_id.vat_label on 'Tax ID'"/><span t-field="doc.partner_id.vat"/></p>
10 </div>
11 <t-if="doc.partner_shipping_id == doc.partner_invoice_id
12 and doc.partner_invoice_id != doc.partner_id
13 or doc.partner_shipping_id != doc.partner_invoice_id">
14 <t-set="information_block">
15 <strong t-if="doc.partner_shipping_id == doc.partner_invoice_id">Invoicing and Shipping Address:</strong>
16 <strong t-if="doc.partner_shipping_id != doc.partner_invoice_id">Invoicing Address:</strong>
17 <div t-field="doc.partner_invoice_id">
18 <t-options widget="contact", fields=["address", "name", "phone"], no_marker=True, phone_icons=True />
19 <t-if="doc.partner_shipping_id != doc.partner_invoice_id">
20 <strong Shipping Address:</strong>
21 <div t-field="doc.partner_shipping_id">
22 <t-options widget="contact", fields=["address", "name", "phone"], no_marker=True, phone_icons=True />
23 </div>
24 </div>
25 </t>
26 </div class="page">
27 <div class="oe_structure">
28 <h2 class="mt16">
29 <t-if="not (env.context.get('proforma', False) or is_proforma)">
30 <span t-if="doc.state not in ['draft', 'sent']">Order # </span>
31 <span t-if="doc.state in ['draft', 'sent']">Quotation # </span>
32 </t>
33 <t-if="env.context.get('proforma', False) or is_proforma">
34 <span>Pro-Forma Invoice # </span>
35 </t>
36 <span t-field="doc.name"/>
37 </h2>
38 <div class="row mt32 mb32" id="informations">
39 <div t-if="doc.client_order_ref" class="col-auto col-3 mw-100 mb-2">
40 <strong>Your Reference:</strong>
41 <p class="m-0" t-field="doc.client_order_ref">
42 </div>
43 <div t-if="doc.date_order and doc.state not in ['draft', 'sent']" class="col-auto col-3 mw-100 mb-2">
44 <strong>Order Date:</strong>
45 <p class="m-0" t-field="doc.date_order">
46 </div>
47 </div>
48 </div>

```

Ilustración 41: Vista 2 "sale_report_template.xml"

Una vez entendido las referencias a las vistas y modelos, definiremos nuestra vista de facturación para el informe personalizado. En nuestro caso nuestro informe personalizado, a nivel de código, se va a llamar "custom_saleorder".

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <data>
4 <record id="action_report_saleorder" model="ir.actions.report">
5 <field name="name">Quotation / Order</field>
6 <field name="model">sale.order</field>
7 <field name="report_type">qweb-pdf</field>
8 <field name="report_name">sale_report_saleorder</field>
9 <field name="report_file">sale_report_saleorder</field>
10 <field name="print_report_name">(object.state in ('draft', 'sent') and 'DiseñAlmeria - %s' % (object.name)) or 'DiseñAlmeria - %s' %
11 <field name="binding_model_id" ref="model_sale_order"/>
12 <field name="binding_type">report</field>
13 </record>
14 <record id="action_report_proforma_invoice" model="ir.actions.report">
15 <field name="name">PRO-FORMA Invoice</field>
16 <field name="model">sale.order</field>
17 <field name="report_type">qweb-pdf</field>
18 <field name="report_name">sale_report_saleorder_proforma</field>
19 <field name="report_file">sale_report_saleorder_proforma</field>
20 <field name="print_report_name">'PRO-FORMA - %s' % (object.name)</field>
21 <field name="binding_model_id" ref="model_sale_order"/>
22 <field name="binding_type">report</field>
23 <field name="groups_id" eval="[(4, ref('sale.group_proforma_sales'))]" />
24 </record>
25 <record id="pedido_de_venta_disenalmeria" model="ir.actions.report">
26 <field name="name">Informe del pedido de venta</field>
27 <field name="model">sale.order</field>
28 <field name="report_type">qweb-pdf</field>
29 <field name="report_name">sale.custom_saleorder</field>
30 <field name="report_file">sale.custom_saleorder</field>
31 <field name="print_report_name">'Sale Order - %s' % (object.name)</field>
32 <field name="binding_model_id" ref="model_sale_order" />
33 <field name="binding_type">report</field>
34 <field name="paperformat_id" ref="sale_order_report_003068_paperformat" />
35 </record>
36 </data>
37 </odoo>

```

Ilustración 42: Report "pedido_de_venta_disenalmeria"

Nos iremos a "sale_report_template" y declaramos nuestra template.

De igual forma que en el anterior, desde “sale_report” llama a nuestro reporte con id “custom_saleorder” y éste hace una llamada a “report_custom_saleorder_document” que se trata del template que contiene todo lo que se va a mostrar por pantalla.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <template id="report_saleorder_document">...
210 </template>
211
212
213 <template id="report_saleorder">
214 <t-call="web.html_container">
215 <t-foreach="docs" t-as="doc">
216 <t-call="sale.report_saleorder_document" t-lang="doc.partner_id.lang"/>
217 </t>
218 </t>
219 </template>
220
221 <template id="report_custom_saleorder_document">
222 <t-call="web.external_layout">
223 <t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
224 <div class="page">
225
226 </div>
227 </t>
228 </template>
229
230 <template id="custom_saleorder">
231 <t-call="web.html_container">
232 <t-foreach="docs" t-as="doc">
233 <t-call="sale.report_custom_saleorder_document" t-lang="doc.partner_id.lang"/>
234 </t>
235 </t>
236 </template>
237
238 <template id="report_saleorder_pro_forma">
239 <t-call="web.html_container">
240 <t-set="is_pro_forma" t-value="True"/>
241 <t-set="docs" t-value="docs.with_context(proforma=True)/>
242 <t-foreach="docs" t-as="doc">
243 <t-call="sale.report_saleorder_document" t-lang="doc.partner_id.lang"/>
244 </t>
245 </t>
246 </template>
247 </odoo>
248

```

Ilustración 43: Llamada a reporte.

Con esta llamada, carga el formato, el idioma y la página “page” que primeramente va a estar en blanco.

Podemos observar este nuevo reporte desde el propio módulo de Odo, confirmando su correcta creación.

| Crear | Modelo | Acción | Reporte | Formato |
|--------------------------|------------------------|-------------------|---|---|
| <input type="checkbox"/> | product product | ir.actions.report | product.report_productbarcode | PDF |
| <input type="checkbox"/> | product template | ir.actions.report | product.report_producttemplatebarcode | PDF |
| <input type="checkbox"/> | stock location | ir.actions.report | stock.report_location_barcode | PDF |
| <input type="checkbox"/> | stock picking | ir.actions.report | stock.label_transfer_template_view_pdf | PDF |
| <input type="checkbox"/> | stock quant package | ir.actions.report | stock.report_package_barcode_small | PDF |
| <input type="checkbox"/> | stock picking | ir.actions.report | stock.label_transfer_template_view_zpl | Texto |
| <input type="checkbox"/> | stock quant package | ir.actions.report | stock.label_package_template_view | Texto |
| <input type="checkbox"/> | stock quant package | ir.actions.report | stock.report_package_barcode | PDF |
| <input type="checkbox"/> | product product | ir.actions.report | stock.label_barcode_product_product_view | Texto |
| <input type="checkbox"/> | product template | ir.actions.report | stock.label_barcode_product_template_view | Texto |
| <input type="checkbox"/> | product packaging | ir.actions.report | product.report_packagingbarcode | PDF |
| <input type="checkbox"/> | product template | ir.actions.report | product.report_producttemplabel | PDF |
| <input type="checkbox"/> | product product | ir.actions.report | product.report_productlabel | PDF |
| <input type="checkbox"/> | product template | ir.actions.report | stock.label_product_template_view | Texto |
| <input type="checkbox"/> | product product | ir.actions.report | stock.label_product_product_view | Texto |
| <input type="checkbox"/> | account.bank.statement | ir.actions.report | account.report_statement | PDF |
| <input type="checkbox"/> | sale order | ir.actions.report | sale.report_saleorder_pro_forma | PDF |
| <input type="checkbox"/> | account move | ir.actions.report | account.report_invoice_with_payments | PDF (object state == 'posted') and (object name or IRV) replace('/', '_') |
| <input type="checkbox"/> | account move | ir.actions.report | account.report_original_vendor_bill | PDF (original_vendor_bill.pdf) |
| <input type="checkbox"/> | account move | ir.actions.report | account.report_invoice | PDF (object state == 'posted') and (object name or IRV) replace('/', '_') |
| <input type="checkbox"/> | ir.module.module | ir.actions.report | base.report_immodulereference | PDF |
| <input type="checkbox"/> | stock inventory | ir.actions.report | stock.report_inventory | PDF |
| <input type="checkbox"/> | ir.model | ir.actions.report | base.report_imodeloverview | PDF |
| <input type="checkbox"/> | product product | ir.actions.report | stock.report_product_product_replenishment | HTML |
| <input type="checkbox"/> | product template | ir.actions.report | stock.report_product_template_replenishment | HTML |
| <input type="checkbox"/> | product template | ir.actions.report | stock.report_stock_rule | HTML |
| <input type="checkbox"/> | sale order | ir.actions.report | sale.custom_saleorder | PDF |
| <input type="checkbox"/> | stock production lot | ir.actions.report | stock.report_lot_label | PDF |
| <input type="checkbox"/> | stock production lot | ir.actions.report | stock.label_lot_template_view | Texto |
| <input type="checkbox"/> | stock picking | ir.actions.report | stock.report_picking | PDF |

Ilustración 44: Creación reporte Odo.

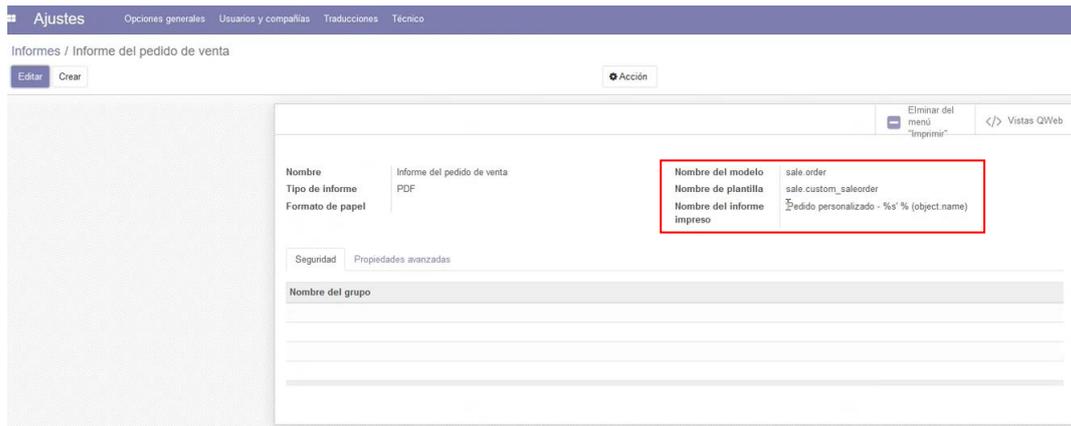


Ilustración 45: Creación reporte Odoo_2.



Ilustración 46: Creación reporte Odoo_3.

Ahora, para sacar nuestro nuevo reporte customizado, para mayor funcionalidad y a petición de la empresa en cuestión, crearemos un botón para que imprima específicamente nuestro reporte.

Para ello, nos iremos al apartado de vistas del módulo de ventas “sale_view” y declararemos nuestro nuevo botón. Éste mostrará el texto “Imprimir pedido de venta”, y su nombre a nivel de código es “action_print_sale”.

```

string="Capture Transaction" class="oe_highlight"
  attrs="{ 'invisible': [(['authorized_transaction_ids', '=', []])]}"/>
<button name="payment_action_void" type="object"
  string="Void Transaction"
  confirm="Are you sure you want to void the authorized transaction? This action can't be undone."
  attrs="{ 'invisible': [(['authorized_transaction_ids', '=', []])]}"/>
<button name="sale.action.view.sale.advance.payment.inv" string="Create Invoice"
  type="action" class="btn-primary"
  attrs="{ 'invisible': [(['invoice_status', '!=', 'no'], ('state', '!=', 'sale'])]}"/>
<button name="sale.action.view.sale.advance.payment.inv" string="Create Invoice"
  type="action" context="{ 'default_advance_payment_method': 'percentage' }"
  attrs="{ 'invisible': [(['invoice_status', '!=', 'no'], ('state', '!=', 'sale'])]}"/>
<button name="action_quotation_send" string="Send by Email" type="object" states="draft" class="btn-primary"/>
<button name="action_quotation_send" type="object" string="Send PRO-FORMA Invoice"
  groups="sale.group.proforma.sales" class="btn-primary"
  attrs="{ 'invisible': [(['state', '!=', 'draft'], ('invoice_count', '>=, 1'])] } context="{ 'proforma': True }"/>
<button name="action_confirm" id="action_confirm"
  string="Confirm" class="btn-primary" type="object"
  attrs="{ 'invisible': [(['state', 'not in', ['sent']])]}"/>
<button name="action_confirm"
  string="Confirm" type="object"
  attrs="{ 'invisible': [(['state', 'not in', ['draft']])]}"/>
<button name="action_quotation_send" type="object" string="Send PRO-FORMA Invoice" groups="sale.group.proforma.sales" attrs="{ 'invisible':
<button name="action_cancel" string="Cancel" type="object" string="Cancel" attrs="{ 'invisible': [(['state', 'not in', ['draft', 'sent', 'sale']])]}"/>
</header>
<sheet>
  <div class="oe_button_box" name="button_box">
    <button name="preview_sale_order"
      type="object"
      class="oe_stat_button"
      icon="fa-globe icon"/>
    <div class="o_field_widget o_stat_info">
      <span class="o_stat_text">Customer</span>
      <span class="o_stat_text">Preview</span>
    </div>
    </div>
    <button name="action_view_invoice"
      type="object"
      class="oe_stat_button"
      icon="fa-pencil-square-o"/>
    <div class="o_stat_info">
      <span class="o_stat_text">[('invoice_count', '=', 0)]>
    </div>
    <field name="invoice_count" widget="statinfo" string="Invoices"/>
  </div>
</sheet>
</div>

```

Ilustración 47: Vista_Botón.

Una vez lo hemos declarado en la vista, nos iremos al modelo de datos de ventas para definir la acción de este botón “def action_print_sale”.

```

for move in moves:
    move.message_post_with_view('mail.message_origin_link',
        values=[self, move, 'origin': move.line_ids.mapped('sale_line_ids_order_id')],
        subtype_id=self.env.ref('mail.mt_note').id
    )
return moves

def action_draft(self):
    orders = self.filtered(lambda s: s.state in ['cancel', 'sent'])
    return orders.write({
        'state': 'draft',
        'signature': False,
        'signed_by': False,
        'signed_on': False,
    })

def action_cancel(self):
    cancel_warning = self._show_cancel_wizard()
    if cancel_warning:
        return {
            'name': _('Cancel Sales Order'),
            'view_mode': 'form',
            'res_model': 'sale.order.cancel',
            'view_id': self.env.ref('sale.sale_order_cancel_view_form').id,
            'type': 'ir.actions.act_window',
            'context': {'default_order_id': self.id},
            'target': 'new'
        }
    return self._action_cancel()

def action_print_sale(self):
    self.ensure_one()
    return self.env.ref("sale.pedido_de_venta_disenalmerid").report_action(
        self
    )

def _action_cancel(self):
    inv = self.invoice_ids.filtered(lambda inv: inv.state == 'draft')
    inv.button_cancel()
    return self.write({'state': 'cancel'})

def _show_cancel_wizard(self):
    for order in self:
        if order.invoice_ids.filtered(lambda inv: inv.state == 'draft') and not order.context.get('disable_cancel_warning'):
            return True
    return False

```

Ilustración 48: Modelo Botón

Este botón lo que hará será llamar al reporte que habíamos declarado previamente.

Cada vez que se trabaja con modelo de datos, es necesario reiniciar el servidor para poder observar los cambios. Una vez reiniciado el servidor, será necesario también reiniciar el módulo donde estén las vistas que vamos a emplear para visualizar los cambios.

Una vez reiniciado, podemos observar los cambios realizados y que se ha creado el botón que habíamos declarado.

Inicialmente, este documento nos mostrará un documento vacío.

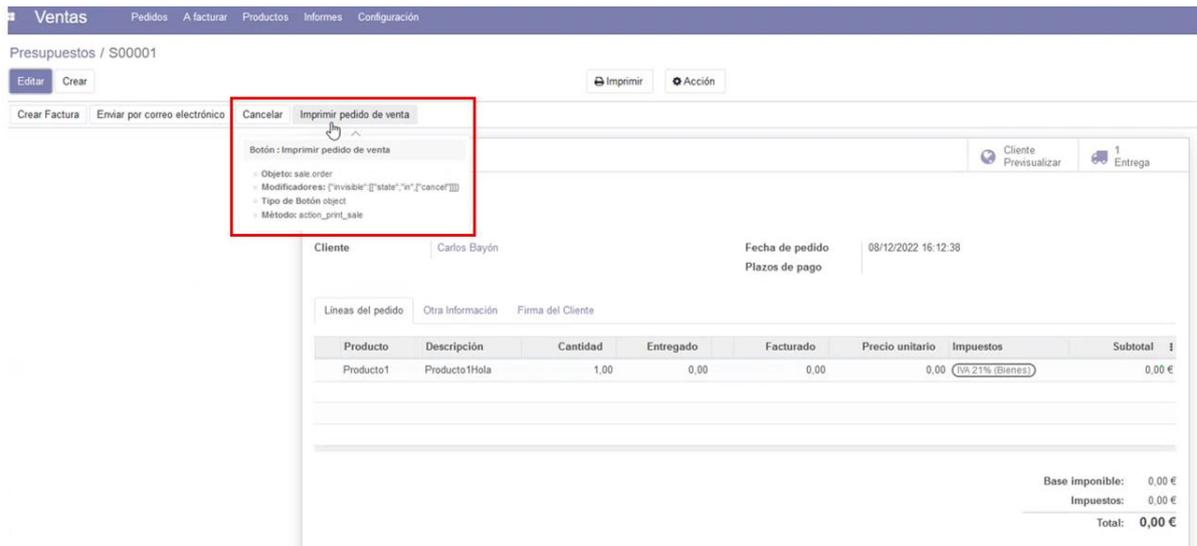


Ilustración 49: Botón Odoo.

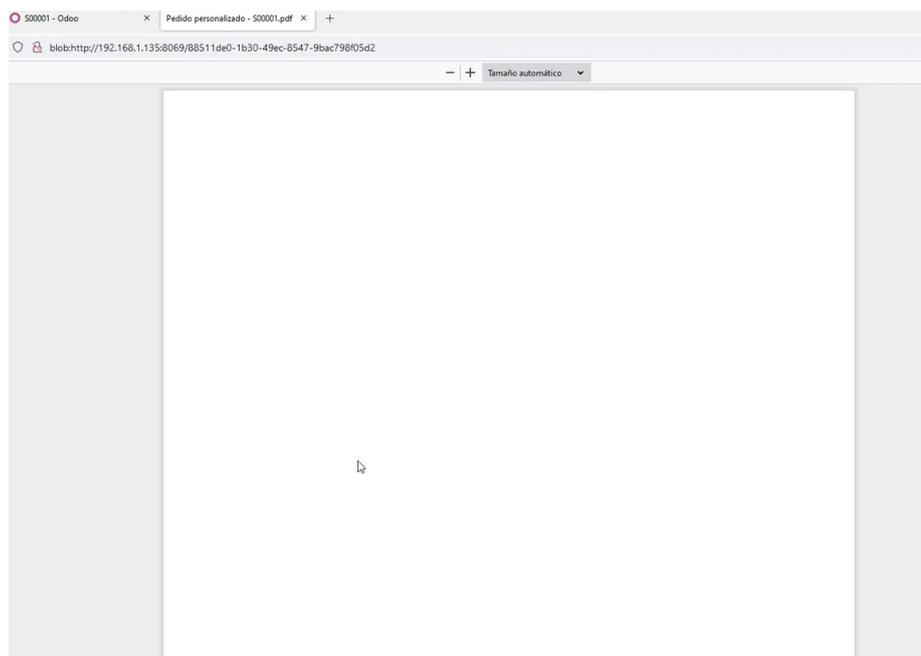


Ilustración 50: Reporte vacío botón

Ahora pasamos a crear nuestro nuevo informe.

Los informes están divididos en 3 partes: la cabecera, la página y el pie de página.

Lo definiremos en nuestro modelo tal que “header” es la cabecera, “page” es la página y “footer” es el pie de página:

Ilustración 51: Vista "report_custom_saleorder"

Ahora tendremos que declarar un diseño del papel donde definiremos algunas de las características principales, como son los márgenes.

Lo que declararemos es que el “pedido_de_venta_disenalmeria” tendrá un formato de papel correspondiente al “disenalmeria_paperformat”

```

report > sale_report.xml
  <field name="binding_type">report</field>
</record>
<record id="action_report_proforma_invoice" model="ir.actions.report">
  <field name="name">PRO-FORMA Invoices</field>
  <field name="model">sale.order</field>
  <field name="report_type">qweb-pdf</field>
  <field name="report_name">sale.report_saleorder_proforma</field>
  <field name="report_file">sale.report_saleorder_proforma</field>
  <field name="print_report_name">PRO-FORMA - %s' % (object.name)</field>
  <field name="binding_model_id" ref="model_sale_order"/>
  <field name="binding_type">report</field>
  <field name="groups_id" eval="[[4, ref('sale.group_proforma_sales')]]"/>
</record>
<record id="disenalmeria_paperformat" model="report.paperformat">
  <field name="name">Disenalmeria Pedido de venta A4</field>
  <field name="default" eval="True" />
  <field name="format">A4</field>
  <field name="page_height">0</field>
  <field name="page_width">0</field>
  <field name="orientation">Portrait</field>
  <field name="margin_top">60</field>
  <field name="margin_bottom">35</field>
  <field name="margin_left">0</field>
  <field name="margin_right">0</field>
  <field name="header_line" eval="False" />
  <field name="header_spacing">60</field>
  <field name="dpi">90</field>
</record>
<record id="pedido_de_venta_disenalmeria" model="ir.actions.report">
  <field name="name">Informe del pedido de venta</field>
  <field name="model">sale.order</field>
  <field name="report_type">qweb-pdf</field>
  <field name="report_name">sale.custom_saleorder</field>
  <field name="report_file">sale.custom_saleorder</field>
  <field name="print_report_name">Pedido de venta - Disenalmeria - Nº %s' % (object.name)</field>
  <field name="binding_model_id" ref="model_sale_order" />
  <field name="binding_type">report</field>
  <field name="paperformat_id" ref="disenalmeria_paperformat" />
</record>
</data>
</odoo>

```

Ilustración 52: Vista formato de papel "disenalmeria"

Ahora, volveremos al modelo de nuestro reporte custom y empezaremos a editar nuestro reporte.

Para un primer lienzo, añadiremos una cabecera con una imagen inicial y delimitaremos las 3 partes de la página.

```

report > sale_report_templates.xml
  <xml version="1.0" encoding="utf-8"?>
  <odoo>
  <template id="report_saleorder_document">...
  </template>
  <template id="report_saleorder">...
  </template>
  <template id="report_custom_saleorder_document">
  <t-call="web.basic_layout">
  <t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
  <div class="header">
  <div class="row">
  
  <h1>miguel</h1>
  </div>
  </div>
  <div class="page">
  <span>
  <h1>holaaaaaaaa</h1>
  </span>
  </div>
  <div class="footer">
  <span>
  <h1>buenassssss</h1>
  </span>
  </div>
  </t>
  </template>
  <template id="custom_saleorder">...
  </template>
  <template id="report_saleorder_proforma">...
  </template>
  </odoo>

```

Ilustración 53: "Límites de cabeceras"

Y si vamos a generar el reporte, vemos que la imagen que nos muestra es la siguiente

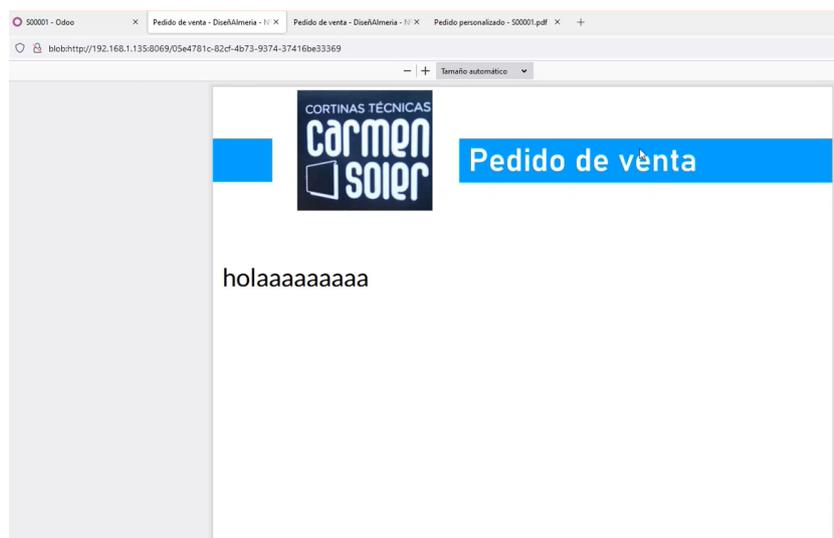


Ilustración 54: Limites de cabeceras_1

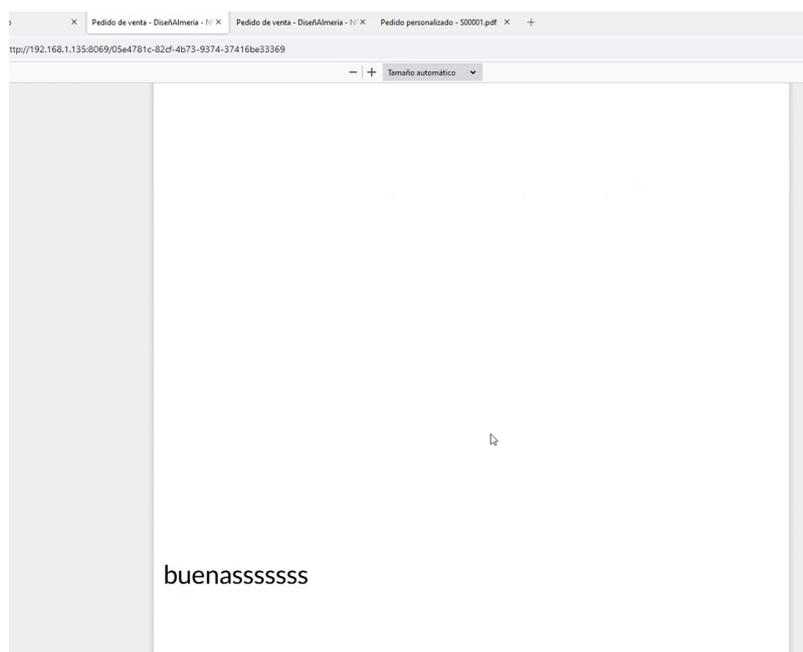


Ilustración 55: Limites de cabeceras_2

Ahora ya tenemos definido donde se sitúan cada parte de nuestro documento según lo que hemos definido en el formato de hoja.

La idea es crear un formato de factura como el siguiente boceto:



Ilustración 56: Boceto Reporte

Para ello, tendremos que ir definiendo las partes del documento según las necesidades.

Empezaremos por la cabecera. Para esta parte, el trabajo es bien sencillo, pues únicamente nos tendremos que encargar de crear un diseño de cabecera a gusto de la empresa. En nuestro caso, ya tenemos modelado la cabecera para la empresa en cuestión:



Ilustración 57: Diseño Cabecera.

El siguiente paso es definirlo en nuestro programa, para ello nos iremos a la parte correspondiente a la cabecera y definiremos nuestra imagen de cabecera.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <odoo>
3 <template id="report_saleorder_document"> ...
210 </template>
211
212
213 <template id="report_saleorder"> ...
219 </template>
220
221 <template id="report_custom_saleorder_document">
222 <t t-call="web.basic_layout">
223 <t t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
224 <div class="header"> ...
234 </div>
235 <div class="page"> ...
351 </div>
352 <div class="footer"> ...
356 </div>
357 </t>
358 </template>
359

```

Ilustración 58: Definir Cabecera

```

224 <div class="header">
225 <div class="row">
226 <t t-if="doc.state != 'draft'">
227 
228 </t>
229 <t t-if="doc.state == 'draft'">
230 <img width="100%" src="" />
231 </t>
232 </div>
233 </div>
234

```

Ilustración 59: Definir Cabecera_2

En el header, le indicamos la ruta del sistema donde tiene que extraer la foto para la cabecera.

Actualizamos el módulo y sacamos un reporte para ver cómo se queda:

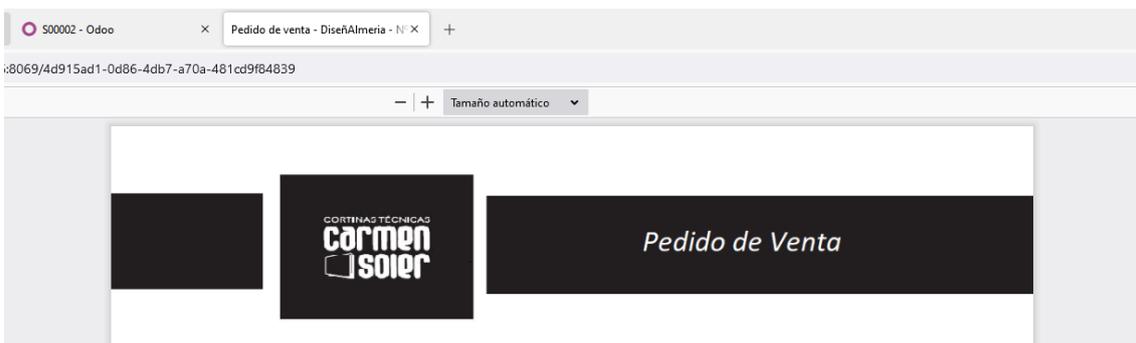


Ilustración 60: Resultado Cabecera.

Una vez hemos configurado la cabecera, el siguiente paso es definir la página.

Según el boceto que habíamos definido, ahora nos encontramos en la parte de la página y tenemos que definir primero las 2 tablas con los datos de la empresa y del cliente.



Ilustración 61: Boceto Reporte_2.

Para ello, nos iremos al fragmento del código correspondiente a la página y definiremos las distintas tablas.

```
sale > report > sale_report_templates.xml
210 </template>
211
212
213 > <template id="report_saleorder">...
219 </template>
220
221 <template id="report_custom_saleorder_document">
222 <t t-call="web.basic_layout">
223 <t t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
224 > <div class="header">...
234 </div>
235 <div class="page">
236 > <table class="table-condensed" style="width:100%;">...
283 </table>
284 <div id="separador_tablas" style="min-height:50px"/>
285 <div style="width: 100%">
286 > <table class="table table-condensed" text-align="center" style="line-height: normal; font-size: 14px;">...
323 </table>
324 </div>
325 <div style="min-height:25px"/>
326 <div class="col-xs-12">
327 <div class="col-xs-8"/>
328 <div class="col-xs-4">
329 > <table class="table table-condensed" width="50%" text-align="right" style="line-height: normal; font-size: 14px;">...
348 </table>
349 </div>
350 </div>
351 </div>
352 > <div class="footer">...
356 </div>
357 </t>
358 </template>
359
360 > <template id="custom_saleorder">...
366 </template>
367
```

Ilustración 62: Configuración "Página Reporte"

El primer fragmento corresponde a la definición de la tabla "table-condensed", donde incorporaremos el código para mostrar la información correspondiente a la parte de cliente como de la empresa.

```

<table class="table-condensed" style="width:100%;">
<tbody>
<tr>
<td>
<span class="text-center"><h3>Datos Compañía</h3></span>
<table id="datos_company" style="width:85%;">
<tbody>
<tr>
<td>
<div style="margin: 1px; border: 2px solid black">
<span>Compañía: <span t-field="doc.company_id.name"/></span><br></div>
<span>Dirección: <span t-field="doc.company_id.partner_id.street"/></span><br></div>
<span>CP: <span t-field="doc.company_id.partner_id.zip"/></span><br></div>
<span>Ciudad: <span t-field="doc.company_id.partner_id.city"/></span><br></div>
<span>País: <span t-field="doc.company_id.partner_id.country_id.name"/></span><br></div>
<span>Teléfono: <span t-field="doc.company_id.phone"/></span><br></div>
<span>NIF: <span t-field="doc.company_id.vat"/></span><br></div>
<span>Email: <span t-field="doc.company_id.email"/></span><br></div>
</td>
</tr>
</tbody>
</table>
</td>
<td>
<span class="text-center"><h3>Datos Cliente</h3></span>
<table id="datos_cliente" style="width:100%;">
<tbody>
<tr>
<td>
<div style="margin: 1px; border: 2px solid black">
<span>Nombre: <span t-field="doc.partner_id.name"/></span><br></div>
<span>Dirección: <span t-field="doc.partner_id.street"/></span><br></div>
<span>CP: <span t-field="doc.partner_id.zip"/></span><br></div>
<span>Ciudad: <span t-field="doc.partner_id.city"/></span><br></div>
<span>País: <span t-field="doc.partner_id.country_id.name"/></span><br></div>
<span>Teléfono: <span t-field="doc.partner_id.phone"/></span><br></div>
<span>NIF: <span t-field="doc.partner_id.vat"/></span><br></div>
<span>Email: <span t-field="doc.partner_id.email"/></span><br></div>
</td>
</tr>
</tbody>
</table>
</td>
</tr>
</tbody>
</table>
<div id="separador_tablas" style="min-height:58px"/>

```

Ilustración 63: Configuración "Página Reporte" 2.

La información de la compañía se recoge en la tabla "datos_company". En ella definimos los campos que queremos mostrar, haciendo referencia a la clase "doc." y los argumentos correspondientes a los de empresa y los de cliente.

Ahora, podemos actualizar la vista y vemos que la imagen que nos muestra es la siguiente:

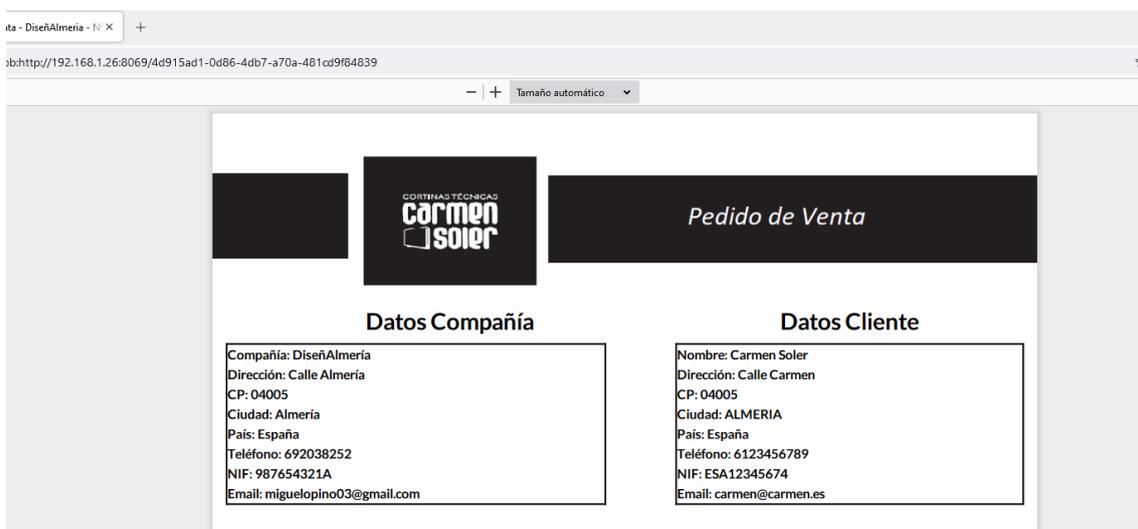


Ilustración 64: Resultado tablas de datos

Lo siguiente es definir la tabla con el listado de productos.



Ilustración 65: Boceto Reporte_2.

Para ello, nos iremos a la siguiente tabla definida en nuestro programa.

```
sale > report > sale_report_templates.xml
210 </template>
211
212
213 > <template id="report_saleorder">...
219 </template>
220
221 <template id="report_custom_saleorder_document">
222 <t t-call="web.basic_layout">
223 <t t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
224 > <div class="header">...
234 </div>
235 <div class="page">
236 > <table class="table-condensed" style="width:100%;">...
283 </table>
284 <div id="separador_tablas" style="min-height:50px"/>
285 <div style="width:100%">
286 > <table class="table table-condensed" text-align="center" style="line-height: normal; font-size: 14px;">...
323 </table>
324 </div>
325 <div style="min-height:25px />
326 <div class="col-xs-12">
327 <div class="col-xs-8"/>
328 <div class="col-xs-4">
329 > <table class="table table-condensed" width="50%" text-align="right" style="line-height: normal; font-size: 14px;">...
348 </table>
349 </div>
350 </div>
351 </div>
352 > <div class="footer">...
356 </div>
357 </t>
358 </template>
359
360 > <template id="custom_saleorder">...
366 </template>
367
```

Ilustración 66: Tabla Productos_1.

```
224 <div class="header">...
225 </div>
226 <table class="table-condensed" style="width:100%;>...
227 </table>
228 <div id="separador_tablas" style="min-height:50px"/>
229 </div style="width:100%">
230 <table class="table table-condensed" text-align="center" style="line-height: normal; font-size: 14px;"
231 <thead id="cabecera_tabla_productos">
232 <tr>
233 <th class="text-center" style="background-color: #B1B1B1">Referencia</th>
234 <th class="text-center" style="background-color: #B1B1B1">Producto</th>
235 <th class="text-center" style="background-color: #B1B1B1">Descripción</th>
236 <th class="text-center" style="background-color: #B1B1B1">Cantidad</th>
237 <th class="text-center" style="background-color: #B1B1B1">Precio</th>
238 <th class="text-center" style="background-color: #B1B1B1">Impuesto</th>
239 <th class="text-center" style="background-color: #B1B1B1">Subtotal</th>
240 </tr>
241 </thead>
242 <tbody id="cuerpo_tabla_productos" text-align="center">
243 <tr t-foreach="doc_order_line" t-as="linea_de_pedido">
244 <td class="text-center">
245 <span t-field="linea_de_pedido.product_id.default_code"/>
246 </td>
247 <td class="text-center">
248 <span t-field="linea_de_pedido.product_id.name"/>
249 </td>
250 <td class="text-center">
251 <span t-field="linea_de_pedido.name"/>
252 </td>
253 <td class="text-center">
254 <span t-field="linea_de_pedido.product_uom_qty"/>
255 </td>
256 <td class="text-center">
257 <span t-field="linea_de_pedido.price_unit"/>
258 </td>
259 <td class="text-center">
260 <span t-field="linea_de_pedido.tax_id.name"/>
261 </td>
262 <td class="text-center">
263 <span t-field="linea_de_pedido.price_subtotal"/>
264 </td>
265 </tr>
266 </tbody>
267 </table>
268 </div>
269 <div style="min-height:25px"/>
270 <div class="col-xs-12">
271 <div class="col-xs-8">
```

Ilustración 67: Tabla Productos_2.

En este fragmento de código es donde definimos la tabla con el listado de productos.

Lo primero es definir la cabecera del listado (thead id="cabecera_tabla_productos"), en ella indicaremos las distintas columnas que tendremos y que se mostrarán, correspondientes a cada producto. Para nuestro reporte hemos decidido mostrar 7 campos diferentes.

El siguiente fragmento (tbody id="cuerpo_tabla_productos") corresponde a la asignación del valor a los distintos campos que hemos definido en nuestra cabecera.

Dado que el listado será dinámico, vemos que contamos con un "t-foreach" que hace referencia a cada orden del documento. Para esta referencia dinámica, se le asigna el nombre línea de pedido (t-as="linea_de_pedido"). Este nombre es el que emplearemos para la asignación de los valores que completarán en los campos de nuestro listado.

Un ejemplo sencillo, es que por cada producto nos muestre su nombre de producto. En nuestro código, este apartado corresponde al 2º elemento de la cabecera y en el cuerpo a la asignación de "línea_de_pedido.product_id.name".

Actualizamos el módulo para ver los cambios e introducimos algunos ejemplos para ver el resultado:

| Referencia | Producto | Descripción | Cantidad | Precio | Impuesto | Subtotal |
|------------|-----------|-------------------|----------|--------|------------------|----------|
| 12345 | Producto1 | [12345] Producto1 | 1,00 | 5,00 | IVA 21% (Bienes) | 5,00 € |
| 12346 | Producto2 | [12346] Producto2 | 2,00 | 10,00 | IVA 21% (Bienes) | 20,00 € |
| 12347 | Producto3 | [12347] Producto3 | 5,00 | 50,00 | IVA 21% (Bienes) | 250,00 € |

Ilustración 68: Resultado Tabla Productos.

El siguiente paso es la tabla de totales según nuestro boceto:



Ilustración 69: Boceto Reporte_3.

Nos iremos al fragmento de código de la página y crearemos la última tabla:

```

210 </template>
211
212
213 > <template id="report_saleorder"> ...
219 </template>
220
221 <template id="report_custom_saleorder_document">
222 <t t-call="web.basic_layout">
223 <t t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
224 > <div class="header"> ...
234 </div>
235 <div class="page">
236 > <table class="table-condensed" style="width:100%;"> ...
283 </table>
284 <div id="separador_tablas" style="min-height:50px"/>
285 <div style="width: 100%">
286 > <table class="table table-condensed" text-align="center" style="line-height: normal; font-size: 14px;"> ...
322 </table>
323 </div>
324 <div style="min-height:25px"/>
325 <div class="col-xs-12">
326 <div class="col-xs-8"/>
327 <div class="col-xs-4">
328 > <table class="table table-condensed" width="50%" text-align="right" style="line-height: normal; font-size: 14px;"> ...
343 </table>
344 </div>
345 </div>
351 </div>
352 > <div class="footer"> ...
356 </div>
357 </t>
358 </template>
359
360 > <template id="custom_saleorder"> ...
366 </template>
367

```

Ilustración 70: Configuración Tabla Totales_1.

```

<div style="min-height:25px"/>
<div class="col-xs-12">
<div class="col-xs-8"/>
<div class="col-xs-4">
<table class="table table-condensed" width="50%" text-align="right" style="line-height: normal; font-size: 14px;">
<thead>
<tr>
<th class="text-center" style="background-color: #B1B1B1">Base imponible</th>
<th class="text-center" style="background-color: #B1B1B1">Impuestos</th>
<th class="text-center" style="background-color: #B1B1B1">Total</th>
</tr>
</thead>
<tbody>
<tr>
<td class="text-center">
<span t-field="doc.amount_untaxed"/>
</td>
<td class="text-center">
<span t-field="doc.amount_tax"/>
</td>
<td class="text-center">
<span t-field="doc.amount_total"/>
</td>
</tr>
</tbody>
</table>
</div>
</div>
<div class="footer"> ...
</div>
</t>

```

Ilustración 71: Configuración Tabla Totales_1.

En este fragmento, al igual que el apartado anterior, tendremos que crear una cabecera para nuestra tabla de totales. La tabla “table - condensed”, queremos situarla en el margen derecho del documento, por ello indicaremos que esté alineada a la derecha con “text-align= ”right” ”.

Para rellenar la cabecera abrimos el <thead> e indicaremos los campos a mostrar, que serán los campos “Base Imponible”, “Impuestos” y “Total”.

En el siguiente fragmento abriremos el <tbody> donde asignaremos el valor para cada campo de la tabla. Haremos referencia a los valores que están registrados en el documento y su extensión. En el caso del total, para el último elemento le asignaremos “doc.amount_total”.

Actualizamos el módulo y vemos como queda reflejado según el pedido anterior.

| Referencia | Producto | Descripción | Cantidad | Precio | Impuesto | Subtotal |
|------------|-----------|-------------------|----------|--------|------------------|----------|
| 12345 | Producto1 | [12345] Producto1 | 1,00 | 5,00 | IVA 21% (Bienes) | 5,00 € |
| 12346 | Producto2 | [12346] Producto2 | 2,00 | 10,00 | IVA 21% (Bienes) | 20,00 € |
| 12347 | Producto3 | [12347] Producto3 | 5,00 | 50,00 | IVA 21% (Bienes) | 250,00 € |

| Base imponible | Impuestos | Total |
|----------------|-----------|----------|
| 275,00 € | 57,75 € | 332,75 € |

Ilustración 72: Resultado Tabla Totales.

Y, por último, queda establecer el pie de página de nuestro documento.



Ilustración 73: Boceto Reporte_4.

Esta tarea es muy similar a la de la cabecera, pues el pie de página será el mismo para todos los documentos y no habrá que diferenciar qué imagen escoger según el tipo de reporte como ocurre con la cabecera.

Primero crearemos un diseño de pie de página, que en nuestro caso ya lo hemos diseñado según lo acordado con la entidad. Además, este pie de página debe recoger el texto obligatorio de tratamiento de datos, siendo un diseño tal que así:



Ilustración 74: Diseño Pie de Página.

Nos iremos al fragmento de código correspondiente al pie de página, en nuestro caso “footer” e indicaremos el pie de página que deberá emplear.

```

sale_report_templates.xml | pedidodeventacabecera.png | sale.py | sale_report.xml | Workspace Trust
sale > report > sale_report_templates.xml
210 </template>
211
212
213 > <template id="report_saleorder"> ...
219 </template>
220
221 <template id="report_custom_saleorder_document">
222 <t t-call="web.basic_layout">
223 <t t-set="doc" t-value="doc.with_context(lang=doc.partner_id.lang)" />
224 > <div class="header">...
234 </div>
235 <div class="page">
236 > <table class="table-condensed" style="width:100%;">...
283 </table>
284 <div id="separador_tablas" style="min-height:50px"/>
285 <div style="width: 100%">
286 > <table class="table table-condensed" text-align="center" style="line-height: normal; font-size: 14px;">...
323 </table>
324 </div>
325 <div style="min-height:25px"/>
326 <div class="col-xs-12">
327 <div class="col-xs-8"/>
328 <div class="col-xs-4">
329 > <table class="table table-condensed" width="50%" text-align="right" style="line-height: normal; font-size: 14px;">...
348 </table>
349 </div>
350 </div>
351 </div>
352 > <div class="footer">...
356 </div>
357 </t>
358 </template>
359
360 > <template id="custom_saleorder"> ...
366 </template>
367

```

Ilustración 75: Configuración Pie de Página.

```

</div>
<div class="footer">
  <div class="row">
    
  </div>
</div>
</t>
</template>

```

Ilustración 76: Configuración Pie de Página_2.

Una vez le indicamos la imagen que cargar, lo siguiente ya es actualizar el módulo y ver el resultado final.

RESULTADO FINAL



Pedido de Venta

Datos Compañía

Compañía: DiseñAlmería
Dirección: Calle Almería
CP: 04005
Ciudad: Almería
País: España
Teléfono: 692038252
NIF: 987654321A
Email: miguelopino03@gmail.com

Datos Cliente

Nombre: Carmen Soler
Dirección: Calle Carmen
CP: 04005
Ciudad: ALMERIA
País: España
Teléfono: 6123456789
NIF: ESA12345674
Email: carmen@carmen.es

| Referencia | Producto | Descripción | Cantidad | Precio | Impuesto | Subtotal |
|------------|-----------|-------------------|----------|--------|------------------|----------|
| 12345 | Producto1 | [12345] Producto1 | 1,00 | 5,00 | IVA 21% (Bienes) | 5,00 € |
| 12346 | Producto2 | [12346] Producto2 | 2,00 | 10,00 | IVA 21% (Bienes) | 20,00 € |
| 12347 | Producto3 | [12347] Producto3 | 5,00 | 50,00 | IVA 21% (Bienes) | 250,00 € |

| Base imponible | Impuestos | Total |
|----------------|-----------|----------|
| 275,00 € | 57,75 € | 332,75 € |



En vista del cumplimiento de la normativa europea 2016/679 sobre Protección de datos (RGPD) le informamos que el tratamiento de los datos proporcionados por Ud. será responsabilidad de CORTINAS TÉCNICAS CARMEN SOLER con el objetivo de Tratamiento Privado de información, y que además es compromete a no ceder o comunicar la información a terceros. Puede ejercer sus derechos de acceso, rectificación, cancelación o supresión del tratamiento a través de csolerdecoracion@gmail.com

689 889 802
csolerdecoracion@gmail.com
[carmen_soler_tecnic](https://www.instagram.com/carmen_soler_tecnic)

Ilustración 77: Resultado Reporte Final

7.2 Configuración Textil:

Por la particularidad de la tipología de empresa, es necesario la personalización de varias partes del ERP, como por ejemplo la personalización de productos. En nuestro caso, personalizando las dimensiones y creando productos particulares a cada cliente.

Nuestra empresa trabaja para diseño tanto interior como exterior, y en su operativa diaria necesitan tomar mediciones para el diseño de un estor, un toldo o lamas para una persiana. Como habíamos tratado anteriormente, en el punto de Gestión de Ventas, requieren de poder realizar facturas de forma dinámica ya que suelen negociar con los proveedores el precio de los productos a la hora de comprarlos.

Es por ello, que necesitamos añadir una funcionalidad extra para que nuestra factura se adecue a las necesidades de nuestra empresa.

Para ello, necesitaremos poder añadir en la factura que se muestre el ancho y el largo del producto que se va a comprar, ya que es un detalle de facturación necesario.

Primero de todo, empezaremos modificando aspectos de los productos de tela que tendrán este tipo de casuística.

Para ello, nos iremos al modelo de datos del producto (“producto_template.py”) y crearemos estos atributos para poder emplearlos más adelante.

```

product_template.py x product_views.xml res_partner.py res_partner_views.xml sale_report_templates.xml sale_report.xml
sale > models > product_template.py
31     default='no',
32     help="Expenses and vendor bills can be re-invoiced to a customer."
33     "With this option, a validated expense can be re-invoice to a customer at its cost or sales price.")
34     visible_expense_policy = fields.Boolean("Re-Invoice Policy visible", compute='_compute_visible_expense_policy', default=lambda self: self._default_visible_expense_policy)
35     sales_count = fields.Float(compute='_compute_sales_count', string='Sold')
36     visible_qty_configurator = fields.Boolean("Quantity visible in configurator", compute='_compute_visible_qty_configurator')
37     invoice_policy = fields.Selection([
38         ('order', 'Ordered quantities'),
39         ('delivery', 'Delivered quantities')], string='Invoicing Policy',
40     help="Ordered Quantity: Invoice quantities ordered by the customer.\n"
41         "Delivered Quantity: Invoice quantities delivered to the customer.",
42     default='order')
43     supplier_id = fields.Many2one([
44         'res.partner',
45         string="Proveedor",
46         domain="[('is_supplier','=', True)]"]
47     )
48     price_per_meter = fields.Float(
49         string="Coste/m (€)",
50         compute="_compute_price_per_meter"
51     )
52     cloth_height = fields.Float(
53         string="Largo (m)"
54     )
55     cloth_width = fields.Float(
56         string="Ancho (m)"
57     )
58     total_price = fields.Float(
59         string="Coste total",
60         compute="_compute_total_price"
61     )
62
63     @api.depends('supplier_id')
64     def _compute_price_per_meter(self):
65         for product_template in self:
66             product_template.price_per_meter = product_template.supplier_id.meter_price
67
68     @api.depends('price_per_meter', 'cloth_height', 'cloth_width')
69     def _compute_total_price(self):
70         for product_template in self:
71             product_template.total_price = product_template.cloth_height * product_template.cloth_width * product_template.price_per_meter
72
73     def _compute_visible_qty_configurator(self):
74         for product_template in self:
75             product_template.visible_qty_configurator = True
76

```

Ilustración 78: Configuración parámetros de medida

Como podemos observar, hemos creado en el modelo de datos los campos Precio por metro(price_per_meter), el largo(cloth_height), el ancho(cloth_with) y el coste total del producto(total_price), además, crearemos las funciones para el cálculo del precio por metro y el cálculo del precio total.

Utilizamos el decorador “@api.depends” con unos campos para que la función siempre se reevalúe cuando alguno de los campos especificados cambie de valor, de forma que nos permita actualizar el cálculo si modificamos algún parámetro como ancho, largo o precio.

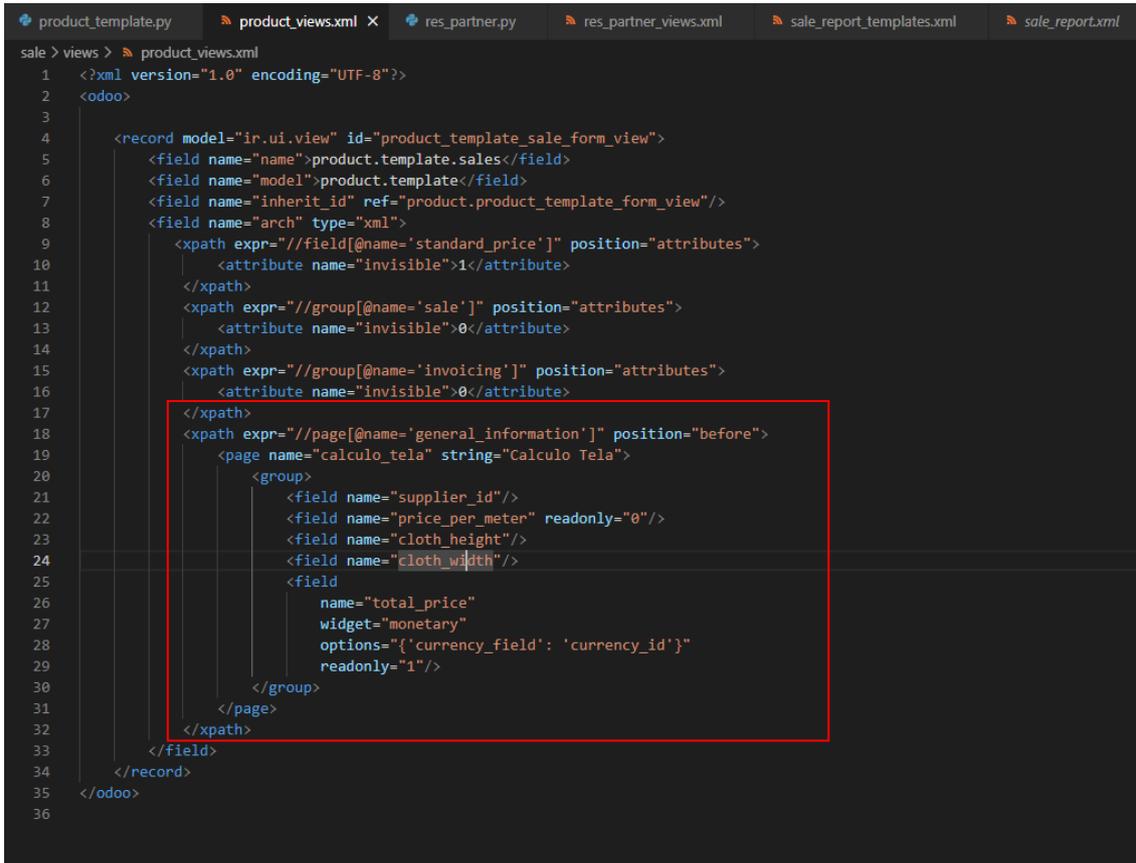
Puesto que es un campo que utiliza varios parámetros para obtener el valor final, añadimos aquellos implicados.

Lo siguiente es crear una pestaña en el propio producto que nos permita poder realizar estas modificaciones sobre el mismo.

Creamos la pestaña de cálculo tela para establecer el proveedor, el coste del material, largo y ancho. Para eso utilizamos el tag “xml <page>”.

Dentro del page, añadiremos una etiqueta de tipo <group> para mostrar los datos de manera agrupada.

Al indicarle `position = "before"`, conseguimos que la primera pestaña que se muestre en la tarjeta del producto sea la del `calculo_tela`.



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <odoo>
3
4 <record model="ir.ui.view" id="product_template_sale_form_view">
5 <field name="name">product.template.sales</field>
6 <field name="model">product.template</field>
7 <field name="inherit_id" ref="product.product_template_form_view"/>
8 <field name="arch" type="xml">
9 <xpath expr="//field[@name='standard_price']" position="attributes">
10 <attribute name="invisible">1</attribute>
11 </xpath>
12 <xpath expr="//group[@name='sale']" position="attributes">
13 <attribute name="invisible">0</attribute>
14 </xpath>
15 <xpath expr="//group[@name='invoicing']" position="attributes">
16 <attribute name="invisible">0</attribute>
17 </xpath>
18 <xpath expr="//page[@name='general_information']" position="before">
19 <page name="calculo_tela" string="Calculo Tela">
20 <group>
21 <field name="supplier_id"/>
22 <field name="price_per_meter" readonly="0"/>
23 <field name="cloth_height"/>
24 <field name="cloth_width"/>
25 <field
26 <name="total_price"
27 <widget="monetary"
28 <options="{ 'currency_field': 'currency_id' }"
29 <readonly="1"/>
30 </group>
31 </page>
32 </xpath>
33 </field>
34 </record>
35 </odoo>
36
```

Ilustración 79: Configuración de vista de parámetros de medida.

Además, podemos apreciar que para el campo `total_price` lo vamos a configurar para que el precio total esté en la divisa establecida por el programa.

De forma que la vista del producto nos queda tal que así:

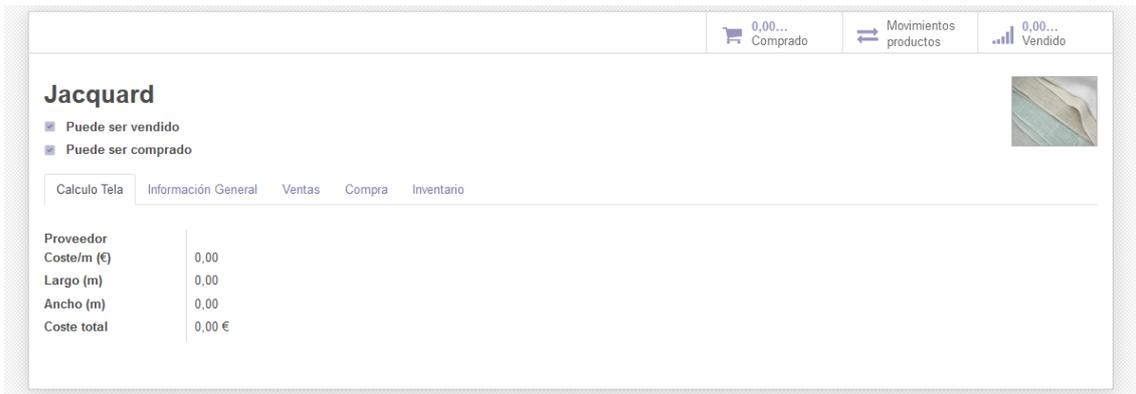


Ilustración 80: Producto Tela Jacquard

Ya hemos casi configurado nuestra calculadora de tela. Como habíamos mencionado anteriormente, la tela la puede comprar a distintos proveedores, por lo que el proveedor lo puede escoger según criterio.

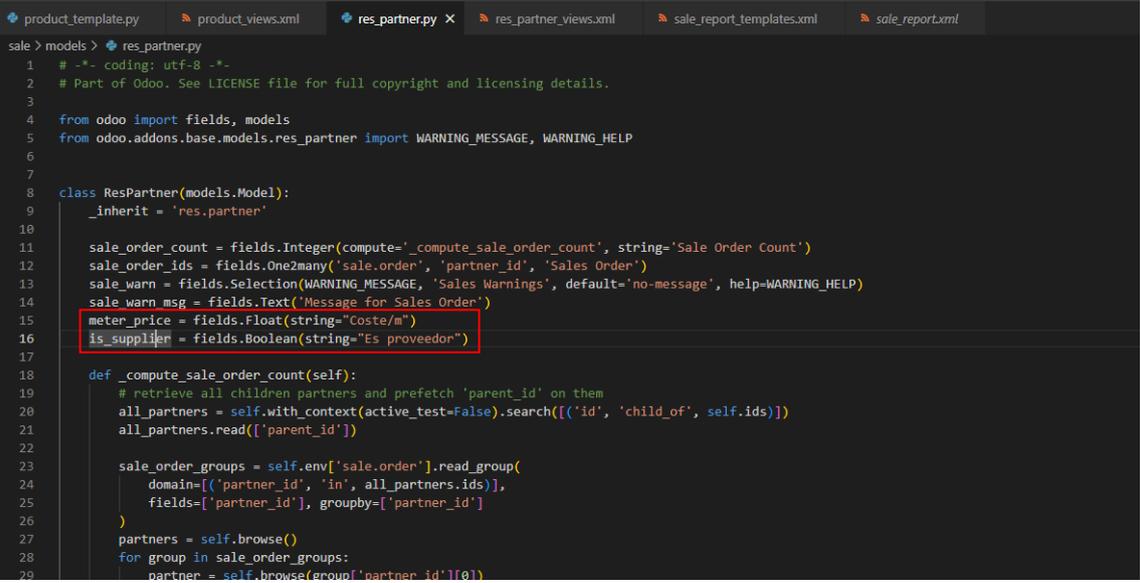
Volviendo al modelo de datos de producto (producto_template.py), apreciamos que hemos creado además el campo “is_supplier”.

```
product_template.py x product_views.xml res_partner.py res_partner_views.xml sale_report_templates.xml sale_report.xml
sale > models > product_template.py
31     default='no',
32     help="Expenses and vendor bills can be re-invoiced to a customer."
33     "With this option, a validated expense can be re-invoice to a customer at its cost or sales price.")
34     visible_expense_policy = fields.Boolean("Re-Invoice Policy visible", compute='_compute_visible_expense_policy', default=lambda self: self._default_visible_expense_policy)
35     sales_count = fields.Float(compute='_compute_sales_count', string='Sold')
36     visible_qty_configurator = fields.Boolean("Quantity visible in configurator", compute='_compute_visible_qty_configurator')
37     invoice_policy = fields.Selection([
38         ('order', 'Ordered quantities'),
39         ('delivery', 'Delivered quantities')], string='Invoicing Policy',
40     help='Ordered Quantity: Invoice quantities ordered by the customer.\n'
41     'Delivered Quantity: Invoice quantities delivered to the customer.',
42     default='order')
43     supplier_id = fields.Many2one(
44         'res.partner',
45         string="Proveedor",
46         domain=[('is_supplier','=', True)],
47     )
48     price_per_meter = fields.Float(
49         string="Coste/m (€)",
50         compute="_compute_price_per_meter"
51     )
52     cloth_height = fields.Float(
53         string="Largo (m)"
54     )
55     cloth_width = fields.Float(
56         string="Ancho (m)"
57     )
58     total_price = fields.Float(
59         string="Coste total",
60         compute="_compute_total_price"
61     )
62
63     @api.depends('supplier_id')
64     def _compute_price_per_meter(self):
65         for product_template in self:
66             product_template.price_per_meter = product_template.supplier_id.meter_price
67
68     @api.depends('price_per_meter', 'cloth_height', 'cloth_width')
69     def _compute_total_price(self):
70         for product_template in self:
71             product_template.total_price = product_template.cloth_height * product_template.cloth_width * product_template.price_per_meter
72
73     def _compute_visible_qty_configurator(self):
74         for product_template in self:
75             product_template.visible_qty_configurator = True
76
```

Ilustración 81: Configuración parámetro proveedor.

Este campo lo hemos creado con el objetivo de que, cuando se quiera asignar un proveedor para un tipo de producto, solamente nos muestre aquellos contactos que sean proveedores. Podemos ver que la información la saca del modelo “res.partner.py” y que con el domain=”[(‘is_supplier’, ‘-’, True)]” filtramos por aquellos contactos que tengan la categoría de proveedor.

Nos vamos al modelo “res_partner.py” para ver la configuración:



```
product_template.py product_views.xml res_partner.py X res_partner_views.xml sale_report_templates.xml sale_report.xml
sale > models > res_partner.py
1 # -*- coding: utf-8 -*-
2 # Part of Odoo. See LICENSE file for full copyright and licensing details.
3
4 from odoo import fields, models
5 from odoo.addons.base.models.res_partner import WARNING_MESSAGE, WARNING_HELP
6
7
8 class ResPartner(models.Model):
9     _inherit = 'res.partner'
10
11     sale_order_count = fields.Integer(compute='_compute_sale_order_count', string='Sale Order Count')
12     sale_order_ids = fields.One2many('sale.order', 'partner_id', 'Sales Order')
13     sale_warn = fields.Selection(WARNING_MESSAGE, 'Sales Warnings', default='no-message', help=WARNING_HELP)
14     sale_warn_msg = fields.Text('Message for Sales Order')
15     meter_price = fields.Float(string='Coste/m')
16     is_supplier = fields.Boolean(string='Es proveedor')
17
18     def _compute_sale_order_count(self):
19         # retrieve all children partners and prefetch 'parent id' on them
20         all_partners = self.with_context(active_test=False).search([('id', 'child_of', self.ids)])
21         all_partners.read(['parent_id'])
22
23         sale_order_groups = self.env['sale.order'].read_group(
24             domain=[('partner_id', 'in', all_partners.ids)],
25             fields=['partner_id'], groupby=['partner_id']
26         )
27         partners = self.browse()
28         for group in sale_order_groups:
29             partner = self.browse(group['partner_id'][0])
```

Ilustración 82: Configuración parámetro proveedor_2.

Una vez están creados los atributos en el modelo de datos de “res_partner.py” nos iremos a la vista para poder incorporar estos atributos en la tarjeta de contacto.

```
product_template.py product_views.xml res_partner.py res_partner_views.xml sale_report_templates.xml sale_report.xml
sale > views > res_partner_views.xml
1 <?xml version="1.0"?>
2 <odoo>
3 <record id="act_res_partner_2_sale_order" model="ir.actions.act_window">
4 <field name="name">Quotations and Sales</field>
5 <field name="res_model">sale.order</field>
6 <field name="view_model">tree,form,graph</field>
7 <field name="context">{ 'default_partner_id': active_id}</field>
8 <field name="groups_id" eval="[(6, ref('sales_team.group_sale_salesman'))]" />
9 <field name="help" type="html">
10 <p class="o_view_nocontent_smiling_face">
11 Create a new quotation, the first step of a new sale!
12 </p><p>
13 Once the quotation is confirmed by the customer, it becomes a sales order.<br/> You will be able to create an invoice and collect the payment.
14 </p>
15 </field>
16 </record>
17
18 <!-- Partner kanban view inherite -->
19 <record model="ir.ui.view" id="crm_lead_partner_kanban_view">
20 </record>
21
22 <record id="res_partner_view_buttons" model="ir.ui.view">
23 <field name="name">res.partner.view.buttons</field>
24 <field name="model">res.partner</field>
25 <field name="inherit_id" ref="base.view_partner_form" />
26 <field name="priority" eval="3" />
27 <field name="groups_id" eval="[(6, ref('sales_team.group_sale_salesman'))]" />
28 <field name="arch" type="xml">
29 <xpath expr="//field[@name='property_payment_term_id']" position="after">
30 <field name="is_supplier"/>
31 <field name="meter_price"/>
32 </xpath>
33 <div name="button_box" position="inside">
34 <button class="oe_stat_button" type="object" name="action_view_sale_order"
35 groups="sales_team.group_sale_salesman"
36 icons="fa-usd">
37 <field string="Sales" name="sale_order_count" widget="statinfo"/>
38 </button>
39 </div>
40 <page name="internal_notes" position="inside">
41 <group colspan="2" col="2" groups="sale.group_warning_sale">
42 <separator string="Warning on the Sales Order" colspan="4" />
43 <field name="sale_warn" nolabel="1" />
44 <field name="sale_warn_msg" colspan="3" nolabel="1"
45 attrs="{ 'required': [('sale_warn', '!=', False), ('sale_warn', '!=', 'no-message')], 'invisible': [('sale_warn', 'in', (False, 'no-message'))]}" />
46 </group>
47 </page>
48 </odoo>
49
50
```

Ilustración 83: Configuración parámetro proveedor_3.

Hemos modificado la tarjeta estándar de Odoo, añadiendo el campo de “Es Proveedor” mediante herencia de código, por lo que usamos el tag `<xpath>`. En vez de sobrescribir directamente toda la tarjeta de contacto, con esto conseguimos añadir un campo directamente en la vista.

En este caso, como hemos especificado el atributo `position = "after"`, controlamos que el campo aparezca delante del campo `property_payment_term_id`

Individual Compañía



Texere Soul Fabrics

| | | | |
|------------------|--|----------------------------|---|
| Dirección | <input type="text" value="CALLE ALMERIA"/> <input type="text" value="Calle 2..."/> <input type="text" value="ALMERIA"/> Estado <input type="text" value="04008"/> <input type="text" value="País"/> | Teléfono | <input type="text" value="950987654"/> |
| IVA | <input type="text" value="ESA12345674"/> | Móvil | <input type="text"/> |
| | | Correo electrónico | <input type="text" value="telas@texere.com"/> |
| | | Enlace a página web | <input type="text" value="http://texere.es"/> |
| | | Categorías | <input type="text" value="Etiquetas..."/> |

[Contactos y direcciones](#) [Venta y compra](#) [Facturación / Contabilidad](#) [Notas internas](#)

Ventas

| | |
|-------------------------|-------------------------------------|
| Comercial | <input type="text"/> |
| Equipo de ventas | <input type="text"/> |
| Plazos de pago | <input type="text"/> |
| Es proveedor | <input checked="" type="checkbox"/> |
| Coste/m | <input type="text" value="10,00"/> |

Compra

| | |
|-------------------------------|--------------------------|
| Plazos de pago | <input type="text"/> |
| Recordatorio de recibo | <input type="checkbox"/> |

Ilustración 84: Configuración parámetro proveedor_4.

Como podemos observar, en la tarjeta de contacto podemos editar el contacto para establecer si es un proveedor y además podemos registrar por defecto el precio de venta del proveedor, que en este caso está establecido el proveedor “TEXERE SOUL FABRICS” y con un coste de 10 euros por metro.

Dado que hemos declarado “is_supplier” como booleano, Odoó automáticamente lo configura para mostrarlo como si fuera un botón.

Ahora podemos observar nuestra configuración personalizada del producto, simulando una compra de la tela para cortinas JACQUARD, que inicialmente estaba sin un proveedor establecido ni ningún parámetro.

Ilustración 85: Configuración Tela Jacquard.

Le daremos a Editar y lo primero que editamos es el proveedor, pudiendo observar que solo nos muestra aquellos que están asignados como proveedor.

| | |
|--|--|
| Nombre del producto | Jacquard |
| <input checked="" type="checkbox"/> Puede ser vendido | |
| <input checked="" type="checkbox"/> Puede ser comprado | |
| Calculo Tela | Información General Ventas Compra Inventario |
| Proveedor | Texere Soul Fabrics |
| Coste/m (€) | |
| Largo (m) | Escriba algo... |
| Ancho (m) | 0,00 |
| Coste total | 0,00 € |

Ilustración 86: Configuración Tela Jacquard_2.

Y al seleccionarlo, nos asigna directamente el coste por metro que le habíamos establecido en su tarjeta de contacto.

| | |
|--|--|
| Nombre del producto | Jacquard |
| <input checked="" type="checkbox"/> Puede ser vendido | |
| <input checked="" type="checkbox"/> Puede ser comprado | |
| Calculo Tela | Información General Ventas Compra Inventario |
| Proveedor | Texere Soul Fabrics |
| Coste/m (€) | 10,00 |
| Largo (m) | 0,00 |
| Ancho (m) | 0,00 |
| Coste total | 0,00 € |

Ilustración 87: Configuración Tela Jacquard_3.

Simularemos que vamos a hacer el pedido de una cortina de 2 metros de largo por 3 de ancho y lo asignaremos para que lo calcule directamente.

| Nombre del producto | |
|---|---------------------|
| Jacquard | |
| <input checked="" type="checkbox"/> Puede ser vendido <input checked="" type="checkbox"/> Puede ser comprado | |
| Calculo Tela Información General Ventas Compra Inventario | |
| Proveedor | Texere Soul Fabrics |
| Coste/m (€) | 10,00 |
| Largo (m) | 2,00 |
| Ancho (m) | 3,00 |
| Coste total | 60,00 € |

Ilustración 88: Configuración Tela Jacquard_4.

Como podemos ver, nos sale un coste total de 60 euros para las medidas que hemos insertado.

Y ahora, tras negociar con el proveedor, hemos conseguido que nos haga una rebaja y nos cobre el precio por metro de 10 a 8 euros, por lo que lo modificamos.

| Nombre del producto | |
|---|---------------------|
| Jacquard | |
| <input checked="" type="checkbox"/> Puede ser vendido <input checked="" type="checkbox"/> Puede ser comprado | |
| Calculo Tela Información General Ventas Compra Inventario | |
| Proveedor | Texere Soul Fabrics |
| Coste/m (€) | 8,00 |
| Largo (m) | 2,00 |
| Ancho (m) | 3,00 |
| Coste total | 48,00 € |

Ilustración 87: Configuración Tela Jacquard_5.

Como podemos ver, una vez hemos insertado el precio y le clickamos fuera de la casilla, se actualiza automáticamente el coste total. Esto nos permite suplir la necesidad de nuestra empresa de poder ajustar precios de productos según la circunstancia.

Ahora toca reflejar todo lo creado aquí en nuestra facturación y poder incorporar en la factura los campos de largo y ancho que nos pide nuestra empresa.

Como vimos en el punto anterior, nos iremos a la vista del reporte (“sale_report_templates”) para poder asignar estos campos y así nos lo muestre.

```
</thead>
<tbody id="cuerpo_tabla_productos" text-align="center">
  <tr t-foreach="doc.order_line" t-as="linea_de_pedido">
    <td class="text-center">
      <span t-field="linea_de_pedido.product_id.default_code"/>
    </td>
    <td class="text-center">
      <span t-field="linea_de_pedido.product_id.name"/>
    </td>
    <td class="text-center">
      <span t-field="linea_de_pedido.name"/>
    </td>
    <td class="text-center">
      <span t-field="linea_de_pedido.product_uom_qty"/>
    </td>
    <td class="text-center">
      <t t-if="linea_de_pedido.product_id.supplier_id.meter_price">
        <span t-field="linea_de_pedido.product_id.supplier_id.meter_price"/>
      </t>
      <t t-else="">
        <span t-field="linea_de_pedido.price_unit"/>
      </t>
    </td>
    <td class="text-center">
      <t t-if="linea_de_pedido.product_id.cloth_width">
        <span t-field="linea_de_pedido.product_id.cloth_width"/>
      </t>
      <t t-else="">
        <span></span>
      </t>
    </td>
    <td class="text-center">
      <t t-if="linea_de_pedido.product_id.cloth_height">
        <span t-field="linea_de_pedido.product_id.cloth_height"/>
      </t>
      <t t-else="">
        <span></span>
      </t>
    </td>
  </tr>
</tbody>
```

Ilustración 89: Configuración detalle de factura.

Podemos observar que hemos incorporado una serie de diferencias respecto al punto anterior de facturación.

Lo primero es el caso del precio de la unidad: Ahora hemos modificado este campo para que nos filtre entre aquellos productos que tienen un coste establecido por metro y de un proveedor (“<t t-if = “...”> y <t t-else=“...”>”) como hemos configurado en la tela Jacquard, a diferencia de los que no cuentan con este precio. En función de lo que encuentre, asignará un precio u otro.

El siguiente elemento que modificamos es el del largo de la tela. Para este caso, queremos hacer que, si un producto cuenta con esta característica, lo mostrará, pero si no lo tiene, este espacio queda vacío. Lo mismo ocurre para el ancho de la tela, en caso de contar con este dato lo mostrará por pantalla y si no lo tiene mostrará vacío.

Habiendo terminado de personalizar nuestra factura, veamos como queda finalmente en nuestro modelo para la tela JACQUARD.



Pedido de Venta

Datos Compañía

Compañía: DiseñAlmería
 Dirección: Calle Almería
 CP: 04005
 Ciudad: Almería
 País: España
 Teléfono: 692038252
 NIF: 987654321A
 Email: miguelopino03@gmail.com

Datos Cliente

Nombre: Carmen Soler
 Dirección: Calle Carmen
 CP: 04005
 Ciudad: ALMERIA
 País: España
 Teléfono: 6123456789
 NIF: ESA12345674
 Email: carmen@carmen.es

| Referencia | Producto | Descripción | Cantidad | Precio | Ancho (m) | Largo (m) | Impuesto | Subtotal |
|----------------|----------|------------------|----------|--------|-----------|-----------|------------------|----------|
| 54321 | Jacquard | [54321] Jacquard | 1,00 | 8,0 | 3,0 | 2,0 | IVA 21% (Bienes) | 48,00 € |
| Base imponible | | | | | Impuestos | | Total | |
| 48,00 € | | | | | 10,08 € | | 58,08 € | |

Ilustración 90: Resultado configuración detalle de factura.

Como podemos observar, ya contamos con los campos ancho y largo en nuestro modelo de facturación, como así nos solicita la empresa en cuestión.

8. Conclusiones

8.1 Conclusión:

Tras haber concluido la implantación del ERP en la entidad, es menester realizar una breve valoración sobre el trabajo desarrollado y los objetivos establecidos. En primer lugar, la idea es poder hacer una contextualización a nivel global respecto de los Sistemas de Información y las distintas tipologías existentes, explicando sus aspectos positivos y negativos y además aplicándolo desde perspectivas empresariales.

Una vez contextualizado el Sistema de Información, se ha asimilado al concepto de ERP, explicando a su vez sus principales características y atributos a tener en cuenta a la hora de tener que realizar una estrategia de implantación.

Como siguiente punto, se ha expuesto la empresa en cuestión, describiendo los procesos de la empresa y la problemática, permitiéndonos entender las necesidades de la empresa para poder satisfacerlas con la implantación del ERP.

Una vez desarrollado el contexto de la empresa, se ha realizado una selección del ERP más conveniente del mercado según las necesidades de la empresa. Gracias a las descripciones del apartado anterior, se ha escogido los 3 ERP principales para la selección: Apache, ADempiere y Odoo. Tras haber realizado una matriz de selección, se ha decidido que Odoo cumple con la mayoría de las características necesarias para los procesos de negocio de nuestra empresa una vez esté implantado

Tras elegir el ERP definitivo, se ha procedido a su instalación en una máquina local para posteriormente trasladar la base de datos al servidor definitivo.

Una vez instalado y configurado, se ha procedido a personalizar el modelo de facturación para cumplir las necesidades de nuestra empresa. Para ello, gracias a que Odoo es un ERP de código abierto, se ha empleado Visual Studio como visor de textos de Python y se ha trabajado con los ficheros fuente de Odoo para poder hacer las adaptaciones oportunas, mostrando el progreso y descripción de la personalización a medida que se ha desarrollado.

Como cierre de conclusión, podemos concluir que se han cumplido con los objetivos de este trabajo académico:

- El desarrollo de los conceptos de Sistemas de Información.
- Entender el ERP como una extensión de un Sistema de Información.
- Contextualizar la empresa destino y analizar su situación para la implantación
- Realizar una selección oportuna del ERP según criterio.
- Instalación y configuración del ERP a medida para la entidad, accediendo a los módulos de Odoo y personalizar el código fuente.

8.2 Líneas Futuras:

Tras haber desarrollado este trabajo académico, se ha podido realizar un trabajo de desarrollo e implantación de un ERP para una PYME dedicada al Diseño e Interiorismo.

Este trabajo me ha supuesto una gran satisfacción tras poder haber aplicado los conocimientos adquiridos durante el grado para cubrir la necesidad de una empresa real y cercana de mi círculo, lo cual ha permitido una gran interacción para el correcto desarrollo y alineamiento estratégico para el mismo.

Sin embargo, debido a cuestiones de carga de trabajo y de tiempo por ambas partes, no se ha podido tratar algunos aspectos que pueden ser de gran valor añadido para la empresa en un futuro, que se desarrollarán igualmente.

Estos son:

- La elaboración de un procedimiento formalizado para la gestión de copias de seguridad de la base de datos de Odoo, algo de carácter relevante de cara a una Auditoría de Sistemas y sobre todo como plan de contingencia para la empresa.
- Elaborar una página web adaptada al negocio para que puedan publicitarse en internet, permitiendo la automatización de pedidos y envíos, para algunos productos.
- Realizar y exponer una matriz de segregación de funciones para el uso del ERP, algo de carácter relevante de cara a una Auditoría de Sistemas, evitando el riesgo de posibles gestiones fraudulentas.
- Otro punto para tener en cuenta es el propio mantenimiento del ERP. En este caso, dada la relación con la entidad, el desarrollo del ERP no ha supuesto un plan plenamente formalizado. Sin embargo, más adelante se desarrollará un Procedimiento para la gestión de cambios y desarrollo del ERP para la empresa en cuestión, algo de carácter relevante de cara a una Auditoría de Sistemas.

9. Referencias y Bibliografía

- Informática, T. &. (2019). ERP: ¿Qué es? Características. Proveedores de ERP. Tecnología & Informática.
<https://www.tecnologia-informatica.com/erp-que-es/>
- Real Academia Española (2022) Significado Sistema de Información.
<https://dpej.rae.es/lema/sistema-de-informaci%C3%B3n>
- Paul Beynon- Davies (2022) Introducción a la informática en las organizaciones
https://www.reverte.com/libro/sistemas-de-informacion_89242/
- Josep Carbosí (2011) Sistemas de información en las empresas
https://books.google.es/books/about/Sistemas_de_informaci%C3%B3n_en_la_empresa.html?id=HtVkkJTFGUAC&redir_esc=y
- GoLive(2022) ¿Qué tipos de ERP existen?
<https://www.onegolive.com/erp-tipos-modalidades-ejemplos/>
- SoftwarePara(2022) ¿Qué es un ERP?¿Para qué sirven?
<https://softwarepara.net/que-es-un-erp-significado/>
- Odoo S.A. (2022) Documentación Odoo.
<https://www.odoo.com/documentation/14.0/es/>
- Seidor (2022) ¿Cuáles son las características principales de un ERP?
<https://www.seidor.com/blog-pyme/cuales-son-las-caracteristicas-de-un-erp>
- Odoo S.A. (2022) Vistas y Modelos en Odoo.
<https://www.odoo.com/documentation/15.0/es/applications/productivity/studio/views.html>
- Kumar, K. &Hillegersberg, J.V., 2000.Enterprise resource planning: introduction. Commun.ACM, 43(4), 22-26.

- Oltra Badenes, R. 2012. Sistemas Integrados de Gestión Empresarial. Evolución histórica y tendencias de futuro. Editorial UPV.

- Microsoft (2022) Definición de ERP.

<https://dynamics.microsoft.com/es-es/erp/define-erp/#:~:text=Definici%C3%B3n%20de%20ERP,suministro%2C%20recursos%20humanos%20y%20operaciones.>

- Orton y Marlene (2004) Definición de ERP.

<https://sites.google.com/site/wikiactividades/temas/planificacionderecursoempresarialeserp>

- Apache OFBiz (2022) Página principal

<https://ofbiz.apache.org/>

- ADempiere (2022) Página principal

<http://www.adempiere.net/web/guest/product>

- Odoo S.A. (2022) Página principal

https://www.odoo.com/es_ES

- NanoBytes (2022) Ventajas de Odoo y por qué es perfecto para PYMES

<https://nanobytes.es/blog/blog-nanobytes-1/ventajas-erp-11>

- DATADEC (2022) ERP Libre VS ERP Propietario

<https://www.datadec.es/blog/erp-open-source-vs-erp-cerrado>

- TICARTE (2022) Perspectiva funcional del ERP

<https://www.ticarte.com/contenido/que-son-los-sistemas-de-planificacion-de-recursos-empresariales-erp>

- OBS Business School(20220) Ventajas y Desventajas de ERP

<https://www.obsbusiness.school/blog/ventajas-y-desventajas-de-un-sistema-erp-en-tu-empresa>

- Ibertech(2022) Diferencias entre software libre y OpenSource
<https://www.ibertech.org/software-livre-e-fonte-aberta-e-mesmo/>
- Odoo S.A. (2022) Conceptos Generales
https://www.odoo.com/documentation/14.0/es/applications/productivity/studio/concepts/understanding_general.html
- Odoo S.A.(2022) Cómo personalizar reportes
https://www.odoo.com/documentation/14.0/es/applications/productivity/studio/how_to/reports.html