

Sistemas de Gestión Empresarial

Andreu Sanz Sanz

2
D
A
M

2
3
I
2
4



REPASO EXAMEN

Sistemes de Gestió Empresarial | 2023-2024

Andreu Sanz Sanz

Tabla de continguts

1. ¿Cuáles son las Fases de implantación de un ERP? Explícalas.	2
2. ¿Qué es el paradigma MVC?¿Qué hace cada una de sus partes?.....	3
3. ¿Qué GUI hemos utilizado para visualizar las bases de datos en Odoo?¿Sobre qué tipo de SGBD actúa?	4
4. ¿Para qué sirve Fail2ban?¿Y Google Authenticator?.....	4
5. ¿Para qué sirve el fichero docker-compose.yaml?	4
6. ¿Para qué sirve el fichero de logs?	4
7. ¿Con cuál comando creamos un módulo?	4
8. Con el comando docker compose up, ¿qué sucede?	4
9. ¿Qué es una imagen, contenedor y volumen en Docker? Explícalo basándote en nuestro proyecto de Odoo.....	4
10. ¿Qué encontramos en el archivo __manifest.py?.....	5
11. ¿Cómo visualizamos los campos de un modelo en Odoo?	5
12. ¿Cuándo hay que modificar los permisos?	5
13. De antemano se sabe, que un SPRINT consiste en un conjunto de tareas a realizar para conseguir un “miniobjetivo”, varios SPRINTS conseguirán el objetivo final. ¿Cómo sería la relación?	5
14. Para calcular campos automáticos...¿Qué atributo utilizamos?	5
15. ¿Qué realiza la función isinstance?¿Y la función timedelta?	5
16. ¿Para qué sirve @api.depends?¿Para qué lo utilizamos?	5
17. ¿Cómo fijamos un campo con un valor por defecto?	5
18. ¿Para qué hemos utilizado lo siguiente?	5
19. ¿Qué tipo de relaciones entre modelos hemos utilizado? Pon un ejemplo.	6
20. ¿Qué puedes extraer del siguiente campo?.....	6

TEMA 5: IMPLANTACIÓN DE UN ERP EN LA EMPRESA

1. ¿Cuáles son las Fases de implantación de un ERP? Explícalas.

Iniciación

Estudiar el ámbito del proyecto

- Nos hemos reunido con los diferentes gerentes de cada departamento para organizarnos y redactar el protocolo para el montaje y la fabricación de cada producto, además de las pruebas preventa.
- Redactamos las mejores formas para llevar a cabo una instalación solar y cómo mejorar el rendimiento de la empresa.

Realizar estudio de viabilidad económica, técnica y organizativa

- Hemos buscado la manera más económica para conseguir los materiales necesarios para la instalación, soportes, herramientas, etc.
- Analizamos exhaustivamente qué módulos serían necesarios para la implementación de Odoo en Solterra.
- Redactamos unos protocolos a seguir para tener una organización para la venta de productos y el montaje de ellos.

Determinar nivel de cambio a un nuevo sistema

Pensamos en los posibles cambios para la empresa al adoptar un ERP como Odoo y de qué manera podemos adaptarnos mejor y reorganizar el equipo para ello.

Organizar el proyecto y planificarlo

Redactamos un conjunto de herramientas y recursos que nos serían útiles en la empresa.

Desarrollo

Análisis detallado

Redactamos cómo y dónde realizar el montaje de nuestra instalación del ERP junto con otros posibles programas que nos harán falta en un futuro. Por esa razón, decidimos tener unos componentes suficientemente potentes.

Diseño físico del sistema

Redactamos sobre papel un boceto de dónde podríamos poner nuestro servidor para Odoo y adaptar el entorno para el servidor por temas de la mejor temperatura en verano, polvo, etc., para prevenir el mantenimiento del servidor.

Diseño lógico del sistema

Redactamos un diseño de la red en la empresa, qué IPs se pueden conectar, y dónde se sitúa esa IP haciendo referencia al dispositivo.

Revisión de previsiones

Por último, comprobamos que todo esté correcto y que no tengamos ningún posible fallo en el diseño que luego nos pueda complicar mucho el trabajo.

Implementación

Adquisición del hardware

Compramos el hardware necesario para llevar a cabo el proyecto; no fue mucho ya que la mayoría ya disponíamos de él.

Desarrollo de software

Instalamos Odoo en el servidor y lo configuramos según las necesidades de la empresa.

Plan de pruebas

Realizamos entornos de prueba para poder testear los cambios en el servidor o en el ERP sin necesidad de sufrir las consecuencias si se produjera un error en el proceso.

Documentación

Redactamos todo lo realizado para en un futuro saber todo lo que hemos hecho además de poder compartir esta información con nuevos empleados de la empresa.

Implantación

Plan de implantación

Estamos ejecutando meticulosamente nuestro plan de implantación para garantizar la continuidad del negocio durante la transición. Durante esta fase, estamos concentrados en la personalización para satisfacer las necesidades específicas de nuestro negocio.

Formación

Estamos proporcionando una capacitación integral a nuestros empleados, asegurándonos de que estén familiarizados con los nuevos procesos y capacitados para manejar solicitudes personalizadas para un mejor trabajo en el futuro.

Conversión y migración de datos

Hemos llevado a cabo una migración efectiva de datos históricos hacia Odoo, asegurando la integridad de la información relacionada con productos personalizados.

Test de aceptación

Cualquier problema identificado durante estas pruebas está siendo abordado de inmediato, con un enfoque especial en aquellos relacionados con la personalización.

Producción y Soporte

Operación normal

Hemos logrado una transición completa hacia la operación normal con Odoo.

Soporte

Tenemos un soporte continuo para resolver cualquier problema con Odoo en nuestros empleados.

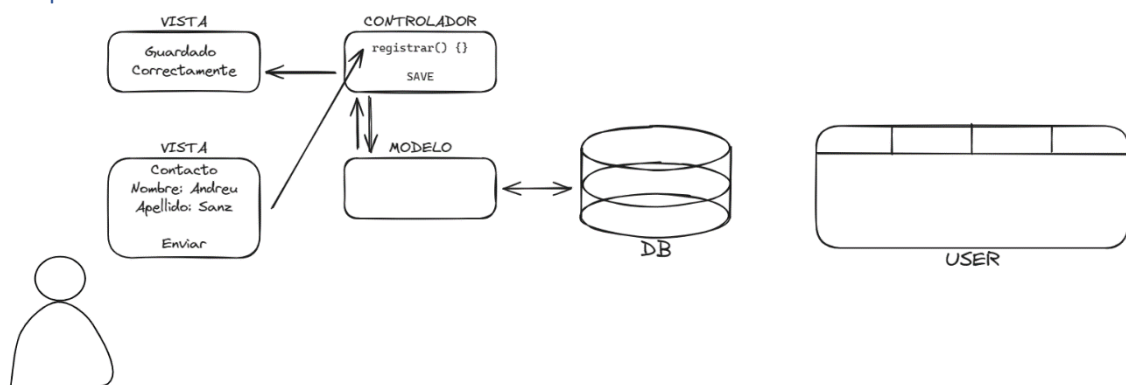
Mantenimiento

Llevamos a cabo una evaluación constante en busca de posibles mejoras en el ERP y la empresa. Aplicamos regularmente actualizaciones y parches de Odoo, garantizando la compatibilidad con las últimas funcionalidades de personalización.

Documentación al cliente

Entrega de documentación actualizada al cliente, destacando los aspectos específicos para el mantenimiento rutinario de su instalación.

Esquema:



2. ¿Qué es el paradigma MVC? ¿Qué hace cada una de sus partes?

El paradigma MVC divide una aplicación en tres partes: el Modelo gestiona datos y lógica de negocio, la Vista maneja la interfaz de usuario y la presentación, y el Controlador coordina la interacción entre el Modelo y la Vista. Esta separación facilita el desarrollo modular, el

mantenimiento del código y la colaboración entre equipos.

3. ¿Qué GUI hemos utilizado para visualizar las bases de datos en Odoo? ¿Sobre qué tipo de SGBD actúa?

PgAdmin, actúa sobre Postgres SQL

4. ¿Para qué sirve Fail2ban? ¿Y Google Authenticator?

Fail2ban es una herramienta que bloquea automáticamente direcciones IP maliciosas basándose en comportamientos detectados en los registros. Google Authenticator proporciona códigos temporales para la autenticación de dos factores, añadiendo una capa extra de seguridad al requerir un segundo factor junto con las credenciales estándar, mejorando la protección contra accesos no autorizados.

TEMA 6: DESARROLLO DE MÓDULOS

5. ¿Para qué sirve el fichero docker-compose.yml?

El archivo docker-compose.yml se utiliza para declarar y configurar servicios, redes y volúmenes en entornos Docker, facilitando la definición y gestión de múltiples contenedores y sus interacciones de manera estructurada.

6. ¿Para qué sirve el fichero de logs?

El archivo de logs registra eventos, errores y actividades cruciales del sistema. Es esencial para la depuración y el diagnóstico de problemas.

7. ¿Con cuál comando creamos un módulo?

Odoo scaffold nombre_modulo

8. Con el comando docker compose up, ¿qué sucede?

El comando `docker-compose up` inicia y crea contenedores según la configuración definida en el archivo `docker-compose.yml`. Este comando también descarga las imágenes necesarias y las construye si es necesario.

9. ¿Qué es una imagen, contenedor y volumen en Docker? Explícalo basándote en nuestro proyecto de Odoo.

Imágenes

Representan plantillas ejecutables que contienen todo lo necesario para ejecutar una aplicación, incluyendo código, bibliotecas y configuraciones, facilitando la implementación consistente de aplicaciones en entornos diversos.

Contenedores

Son instancias ejecutables de imágenes que encapsulan una aplicación y sus dependencias, proporcionando un entorno aislado y portable para su ejecución, con la capacidad de iniciarse, detenerse y moverse fácilmente entre diferentes entornos.

Volúmenes

Son mecanismos para persistir y compartir datos entre contenedores y el sistema host, ofreciendo almacenamiento independiente del ciclo de vida del contenedor, lo que permite manejar datos que requieren persistencia y facilita la colaboración entre múltiples contenedores.

10. ¿Qué encontramos en el archivo `__manifest.py`?

En un archivo `'__manifest__.py'`, se encuentra la información del módulo, como el nombre, la versión, la descripción y la dependencia de otros módulos.

11. ¿Cómo visualizamos los campos de un modelo en Odoo?

Modo desarrollador

Ajuste > Técnico > Modelos

12. ¿Cuándo hay que modificar los permisos?

Debes modificar los permisos cuando necesitas ajustar los niveles de acceso de los usuarios a ciertos objetos, como modelos, vistas o campos en tu aplicación. Nosotros lo hemos modificado cada vez que creábamos un nuevo modelo.

13. De antemano se sabe, que un SPRINT consiste en un conjunto de tareas a realizar para conseguir un “miniobjetivo”, varios SPRINTS conseguirán el objetivo final. ¿Cómo sería la relación?

Sprint : Onetomany

Tasck : Manytoone

14. Para calcular campos automáticos...¿Qué atributo utilizamos?

Para calcular campos automáticos, se utiliza el atributo `'compute'` en la definición del campo. Este atributo señala el método que realizará el cálculo para actualizar el valor del campo de forma automática en función de otros campos o lógica definida en el método especificado.

15. ¿Qué realiza la función `isinstance`? ¿Y la función `timedelta`?

La función `'isinstance'` se utiliza para comprobar si un objeto es una instancia de una clase o de una tupla de clases.

La función `'timedelta'` se utiliza para representar la duración entre dos fechas o tiempos.

16. ¿Para qué sirve `@api.depends`? ¿Para qué lo utilizamos?

Se utiliza para especificar las dependencias entre campos en modelos. Indica qué campos afectan al campo en el que se coloca y, por lo tanto, cuándo se debe actualizar el valor del campo dependiente. En este caso, la función se ejecutará cuando cambien las fechas de inicio (`start_date`) o la duración (`duration`) de un sprint

17. ¿Cómo fijamos un campo con un valor por defecto?

Para fijar un campo con un valor por defecto se utiliza el atributo `'default'`. Este atributo te permite establecer el valor predeterminado que tendrá el campo cuando se cree un nuevo registro en la base de datos.

18. ¿Para qué hemos utilizado lo siguiente?

```
9  import logging
10
11  _logger = logging.getLogger(__name__)
12
```

Para realizar registros (logs) en el módulo actual, proporcionando información útil para el diagnóstico y seguimiento de eventos.

19. ¿Qué tipo de relaciones entre modelos hemos utilizado? Pon un ejemplo.

Many2one:

En la clase task, el campo project establece una relación "many2one" con el modelo manage.project, relacionando las tareas con un proyecto específico.

One2many:

En la clase project, el campo histories establece una relación "one2many" con el modelo manage.history, permitiendo que un proyecto tenga múltiples historias asociadas.

Many2many:

En la clase history, el campo used_technologies establece una relación "many2many" con el modelo manage.technology, registrando las tecnologías utilizadas en todas las tareas asociadas a la historia.

20. ¿Qué puedes extraer del siguiente campo?

```
grupos = fields.One2many(comodel_name='instituto.alumno', inverse_name='alumnos')
```

Este campo 'Grupos' es una relación uno a muchos con el modelo 'instituto.alumno', utilizando el campo inverso 'alumnos'.