**Petitoya**

Manual de Instalación

Versión: 0100

Alpha 0.32.1

**HOJA DE CONTROL**

| **Organismo** | <Nombre Consejería u Organismo Autónomo> | | |
| --- | --- | --- | --- |
| **Proyecto** | Petitoya | | |
| **Entregable** | Manual de Instalación | | |
| **Autor** | <Nombre de la Empresa> | | |
| **Aprobado por** |  | **Fecha Aprobación** | DD/MM/AAAA |
|  |  | **Nº Total de Páginas** |  |

REGISTRO DE CAMBIOS

| **Versión** | **Causa del Cambio** | **Responsable del Cambio** | **Fecha del Cambio** |
| --- | --- | --- | --- |
| 0100 | Versión inicial | <Nombre Apellido1 Apellido2> | DD/MM/AAAA |
|  |  |  |  |
|  |  |  |  |

CONTROL DE DISTRIBUCIÓN

| **Nombre y Apellidos** |
| --- |
| <Nombre Apellido1 Apellido2> |
|  |
|  |
|  |
|  |

[1 . INTRODUCCIÓN 4](#_heading=h.1fob9te)

* 1. [Objeto 4](#_heading=h.3znysh7)0

[1.2 . Alcance 4](#_heading=h.2et92p0)

[2 . DESCRIPCIÓN DEL SISTEMA 5](#_heading=h.1t3h5sf)

[2.1. Antecedentes y descripción funcional del sistema 5](#_heading=h.4d34og8)

[2.2 . Componentes fundamentales 5](#_heading=h.2s8eyo1)

[2.3 . Relación con otros sistemas 5](#_heading=h.17dp8vu)

[3 . RECURSOS HARDWARE 6](#_heading=h.3rdcrjn)

[3.1. Servidores 6](#_heading=h.26in1rg)

[3.2 . Estaciones cliente 6](#_heading=h.lnxbz9)

[3.3 . Conectividad 6](#_heading=h.35nkun2)

[3.4 . Restricciones 7](#_heading=h.1ksv4uv)

[4 . RECURSOS SOFTWARE 8](#_heading=h.44sinio)

[4.1. Matriz de certificación 8](#_heading=h.2jxsxqh)

[4.2 . Restricciones técnicas del sistema 8](#_heading=h.z337ya)

[4.3 . Requisitos de otros sistemas 8](#_heading=h.3j2qqm3)

[5 . INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE 10](#_heading=h.1y810tw)

[6 . CONFIGURACIÓN DEL SISTEMA 11](#_heading=h.4i7ojhp)

[6.1. Configuración del sistema 11](#_heading=h.2xcytpi)

[6.2 . Configuración de otros sistemas 12](#_heading=h.1ci93xb)

[7 . COMPILACIÓN DEL SISTEMA 13](#_heading=h.3whwml4)

[8 . INSTALACIÓN DEL SISTEMA 14](#_heading=h.2bn6wsx)

[8.1. Requisitos previos 14](#_heading=h.qsh70q)

[8.2 . Procedimiento de instalación 14](#_heading=h.3as4poj)

[9 . VERIFICACIÓN DEL PROCESO DE INSTALACIÓN 16](#_heading=h.1pxezwc)

[10. MARCHA ATRÁS DE LA INSTALACIÓN Y CONFIGURACIÓN 17](#_heading=h.49x2ik5)

[10.1. Requisitos previos 17](#_heading=h.2p2csry)

[10.2 . Marcha atrás del sistema 17](#_heading=h.147n2zr)

[10.3 . Marcha atrás del software base 17](#_heading=h.3o7alnk)

[11 . ANEXOS 19](#_heading=h.23ckvvd)

[11.1. Resumen de tareas de configuración 19](#_heading=h.ihv636)

[12 . GLOSARIO 20](#_heading=h.32hioqz)

[13. BIBLIOGRAFÍA Y REFERENCIAS 21](#_heading=h.1hmsyys)

# INTRODUCCIÓN

## Objeto

El propósito de este manual de instalación es ofrecer instrucciones claras y detalladas para garantizar la correcta y eficiente instalación del software, minimizando errores y asegurando el cumplimiento de los procedimientos técnicos establecidos.

## Alcance

Este manual está diseñado para guiar a los usuarios a lo largo de todo el proceso de instalación del software [Nombre del Software]. Cubre desde la descripción del sistema y sus componentes, hasta los procedimientos técnicos necesarios para su correcta instalación y configuración. En detalle, abarca:

1. **Descripción del sistema**: Se proporciona una explicación funcional del software, sus componentes clave y su interacción con otros sistemas.
2. **Recursos necesarios**: Se especifican los requerimientos de hardware y software, así como las restricciones técnicas que deben tenerse en cuenta antes de proceder con la instalación.
3. **Instalación y configuración**: Instrucciones paso a paso para la instalación del software base, la configuración del sistema, y los ajustes requeridos en otros sistemas relacionados.
4. **Compilación y verificación**: Procedimientos para compilar el sistema, verificar el proceso de instalación y asegurar que todo funcione de manera óptima.
5. **Marcha atrás**: Detalles sobre cómo revertir la instalación y la configuración en caso de ser necesario.

Este manual no cubre escenarios de instalación en entornos no soportados ni la personalización avanzada del software. Se centra exclusivamente en la instalación estándar y certificada según las especificaciones del fabricante.

# DESCRIPCIÓN DEL SISTEMA

## Antecedentes y descripción funcional del sistema

El sistema **Petitoya** surge para resolver las ineficiencias en la gestión manual de cafeterías, como la toma de pedidos, el control de inventarios y la gestión de ventas. Estas tareas suelen ser lentas y propensas a errores, lo que afecta la experiencia del cliente y la rentabilidad.

**Petitoya** automatiza estos procesos clave, permitiendo un control preciso de inventarios, una toma de pedidos eficiente y la integración de diversas formas de pago. Además, proporciona estadísticas de ventas y productos, mejorando la toma de decisiones y optimizando el servicio al cliente.

## Componentes fundamentales

| **Módulo** | **Descripción** |
| --- | --- |
| Gestión de Productos | Permite registrar, actualizar y administrar productos, incluyendo categorías y precios. |
| Toma de Pedidos | Facilita la creación y personalización de pedidos, en línea. |
| Informes y Análisis | Genera informes de ventas y análisis, proporcionando estadísticas sobre los productos más y menos vendidos. |
| Administración | Administra usuarios, permisos, y accesos del personal. |

## Relación con otros sistemas

| **Sistema** | **Relación** |
| --- | --- |
| Sistema de Gestión de Inventarios | Petitoya se integra con sistemas de gestión de inventarios para mostrar la disponibilidad de productos, dependiendo de si los pedidos están activos o inactivos. Esto permite a los usuarios tener visibilidad sobre los productos que se pueden ofrecer. |
| Sistemas de Reportes y Análisis | El sistema utiliza herramientas de análisis de datos para generar informes estadísticos sobre las ventas, productos más vendidos y rendimiento del negocio. |
| Sistemas de Comunicación | Petitoya se relaciona con plataformas de comunicación para enviar notificaciones y actualizaciones a los usuarios y clientes sobre el estado de sus pedidos. |

# RECURSOS HARDWARE

## Servidores

| **Servidor 1** | | |
| --- | --- | --- |
| **Dato** | **Valor mínimo** | **Valor recomendado** |
| Procesador | Intel i3 o equivalente | Intel i5 o equivalente |
| Memoria RAM | 8 GB | 16 GB |
| Tamaño Almacenamiento | Unidad SSD de al menos 500 GB | Unidad SSD de al menos 500 GB |
| sistema operativo | Linux (Ubuntu Server) | Linux (Ubuntu Server) |
| internet | Conexión a Internet de alta velocidad. | Conexión a Internet de alta velocidad. |

## Estaciones cliente

| **Dato** | **Valor mínimo** | **Valor recomendado** |
| --- | --- | --- |
| Procesador | intel celeron o superior | intel celeron o superior |
| Memoria RAM | 4 gb | 8 gb |
| Tamaño Almacenamiento | 120 gb | 120 gb |
| navegador | recomendacion usar google chrome | recomendacion usar google chrome |
| internet | Conexión a Internet de alta velocidad. | Conexión a Internet de alta velocidad. |

## Conectividad

| **Dato** | **Valor mínimo** | **Valor recomendado** |
| --- | --- | --- |
| Tarjeta de Red | 10/100 Mbps | 1 Gbps |
| Tipo de Red | Red Local (LAN) con acceso a Internet | Red Local (LAN) con acceso a Internet de alta velocidad |
| energia | Considerar el uso de dispositivos de respaldo de energía (UPS) para asegurar la continuidad del servicio en caso de cortes de energía. |  |
| seguridad | Se recomienda la implementación de un router de calidad que soporte múltiples dispositivos y ofrezca seguridad adicional (firewall integrado, VPN). |  |

## Restricciones

| **Restricción** | **Detalle** |
| --- | --- |
| Entrenamiento y Capacitación | Es esencial que el personal que utilizará **Petitoya** reciba capacitación adecuada. La falta de conocimiento puede llevar a errores en el manejo del sistema. |
| Limitaciones de Hardware | Los servidores y estaciones de trabajo deben cumplir con las especificaciones mínimas para garantizar un rendimiento adecuado del sistema. Hardware inferior puede resultar en tiempos de respuesta lentos. |
| Requerimientos de Conectividad | La falta de una conexión a Internet estable y de alta velocidad puede limitar la funcionalidad del sistema, afectando la capacidad de realizar pedidos en tiempo real. |

# RECURSOS SOFTWARE

## Matriz de certificación

<Introduzca contenido y borre cuadro>

**Stack de Desarrollo**: MERN (MySQL, Express.js, React, Node.js)

* Descripción: El sistema utiliza el stack MERN con MySQL para su desarrollo, lo que garantiza una integración eficiente.

**Compilador**: Vite

* Descripción: Se utiliza Vite como herramienta de construcción y desarrollo para mejorar la experiencia de desarrollo.

**Servidor Web**: Apache 2.4 o superior

* Descripción: El sistema se ejecuta en un servidor Apache, garantizando la disponibilidad y el manejo de solicitudes.

**Contenedor**: Docker

Descripción: El sistema está dockerizado para facilitar la implementación y escalabilidad en diferentes entornos.

**Sistema Operativo**: Windows 10 / Linux Ubuntu 20.0

## Restricciones técnicas del sistema

<Cumplimente tabla y borre cuadro>

| **Elemento** | **Descripción** |
| --- | --- |
| Sistema operativo | Windows 10 / Linux Ubuntu 20.0 |
| Servidor de aplicaciones | Apache 2.4 o superior |
| Servidor de base de datos | MySQL 8.0 |
| Compilador | Vite |
| Contenedor | Docker |

## Requisitos de otros sistemas

| **Requisitos de Otros Sistemas** | **Descripción** | **Versión Compatible** |
| --- | --- | --- |
| Sistema de Gestión de Base de Datos (MySQL) | Base de datos para almacenar y gestionar información relacionada con productos, pedidos y usuarios. | MySQL 5.7 o superior |
| Servidor Web (Apache) | Servidor para alojar la aplicación y gestionar las solicitudes HTTP de los usuarios. | Apache 2.4 o superior |
| Entorno de Contenedores (Docker) | Entorno para desplegar la aplicación y facilitar su implementación. |  |
| Entorno de Desarrollo | Herramientas necesarias para el desarrollo, como Node.js y React. |  |
| Sistema Operativo | Plataforma en la que se ejecuta el sistema. | Windows 10 / Linux Ubuntu 20.04 |

# INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE

| **petitoya** | | |
| --- | --- | --- |
| Descripción | Este sistema está diseñado para mejorar la eficiencia operativa y la experiencia del usuario mediante una interfaz intuitiva y funcionalidades automatizadas. | |
| Localización | se puede alojar en un servidor web configurado con Apache, utilizando Docker para la contenedorización. La base de datos MySQL debe estar configurada y accesible para la aplicación. | |
| Procedimiento de instalación | | |
| Paso 1 | Descargar el código fuente de **Petitoya** desde el repositorio oficial. | |
| Paso 2 | Instalar Docker en el servidor si aún no está instalado. | |
| Paso 3 | Clonar el repositorio en el servidor y navegar a la carpeta del proyecto. | |
| Paso 4 | Construir la imagen de Docker utilizando el comando docker-compose build. | |
| Paso 5 | Iniciar el contenedor con docker-compose up -d. | |
| Paso 6 | Configurar la base de datos MySQL, creando las tablas necesarias utilizando el script de inicialización proporcionado. | |
| Procedimiento de configuración | | |
| Paso 1 | Acceder a la interfaz de administración de **Petitoya**. | |
| Paso 2 | Configurar las credenciales de la base de datos en el archivo de configuración (por ejemplo, en un archivo .env o en un archivo de propiedades). | |
| Paso 4 | Definir los parámetros de la aplicación, como el puerto de conexión y las configuraciones de seguridad. | |
| Paso 5 | Personalizar los productos disponibles en el sistema, habilitando o deshabilitando según sea necesario. | |
| Parámetros que configurar | | |
| **Base de Datos** | [BBDD] | Petitoya |
| **Puerto de Servidor** | [fichero .env] | cambiar a direccion y puerto al que escucha |

# CONFIGURACIÓN DEL SISTEMA

## Configuración del sistema

| **Configuración: docker** | | |
| --- | --- | --- |
| **Efecto** | | **Asegura el correcto funcionamiento de *Petitoya* y permite ejecutar el servidor de backend.** |
| **Fase** | | **Instalación y configuración del entorno de desarrollo.** |
| **Ubicación** | | **Raiz del software** |
| **Paso** | **Descripción** | |
| **1º** | Descargue el repositorio del sofware | |
| **2º** | Ejecute el archivo dockerfil, del fronend y el backend. | |
| **3º** | ejecute el archivo dockecompose. | |

## Configuración de otros sistemas

| **base de datos** | | |
| --- | --- | --- |
| Descripción | La base de datos utilizada por *Petitoya* es MySQL, que almacena toda la información relacionada con productos, pedidos, clientes y usuarios. | |
| Localización | La base de datos se encuentra en el servidor MySQL configurado en un contenedor | |
| Procedimiento de configuración | | |
| Paso 1 | en la carpeta del backen esta al archivo “petitoya.sql” | |
| Paso 2 | importas ese archivo en el mysql | |

# COMPILACIÓN DEL SISTEMA

| **Requisitos de compilación** | |
| --- | --- |
| **Requisito** | **Descripción** |
| **Docker** | Se requiere tener Docker instalado para ejecutar contenedores y gestionar el entorno de desarrollo. |
| **Docker Compose** | Se necesita Docker Compose para definir y ejecutar aplicaciones multicontenedor. |
| **Node.js** | El sistema requiere Node.js para la ejecución del backend y las herramientas de desarrollo. |
| **MySQL** | Base de datos necesaria para almacenar los datos del sistema Petitoya. |
| **Vite** | Herramienta de construcción que se utiliza para compilar la parte del frontend. |

| **Procedimiento de compilación** | |
| --- | --- |
| **Paso** | **Descripción** |
| **1** | abre un terminal en el directorio del proyecto *Petitoya*. |
| **2** | Ejecuta Docker Compose para compilar y levantar los contenedores necesarios. |

# INSTALACIÓN DEL SISTEMA

## Requisitos previos

Antes de proceder con la instalación del sistema Petitoya, asegúrate de cumplir con los siguientes requisitos:

| requisitos | Descripción |
| --- | --- |
| Hardware | Servidor con especificaciones mínimas recomendadas: Intel i5, 16 GB RAM, SSD de 500 GB. |
| software | Sistemas operativos compatibles: Windows 10 / Linux Ubuntu 20.04. Docker y Docker Compose instalados. |
| Red | Acceso a internet de alta velocidad para la descarga de paquetes y actualizaciones. |
| Permisos | Permisos de administrador en el servidor para la instalación de software y configuración del sistema. |

## Procedimiento de instalación

| **Procedimiento de instalación** | |
| --- | --- |
| **Paso 1** | |
| **Tipo** | Preparación del entorno |
| **Componente** | Servidor y software base |
| **Permisos** | Permisos de administrador |
| **Descripción** | Asegúrate de que el servidor cumple con las especificaciones mínimas. Instala el sistema operativo compatible y actualízalo a la última versión disponible. Instala Docker y Docker Compose si aún no están instalados. Configura el acceso a internet. |
| **Paso 2** | |
| **Tipo** | Configuración del entorno |
| **Componente** | Docker y Docker Compose |
| **Permisos** | Permisos de administrador |
| **Descripción** | Descarga e instala Docker y Docker Compose siguiendo las instrucciones del sitio oficial. Verifica la instalación ejecutando docker --version y docker-compose --version en la terminal. |
| **Paso 3** | |
| **Tipo** | Descarga del software |
| **Componente** | Código fuente de Petitoya |
| **Permisos** | Permisos de administrador |
| **Descripción** | Clona el repositorio de Petitoya desde el repositorio oficial utilizando el comando git clone <https://github.com/XxMaxiDiazxX/Petitoya.git> Navega a la carpeta del proyecto clonada. |
| **Paso 4** | |
| **Tipo** | Importación de imágenes de Docker |
| **Componente** | Imágenes de Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Importa las imágenes de Docker necesarias para Petitoya utilizando el comando docker load < [ruta del archivo de imagen]. Asegúrate de que las imágenes se importen correctamente. |
| **Paso 5** | |
| **Tipo** | Construcción y ejecución del contenedor |
| **Componente** | Docker Compose |
| **Permisos** | Permisos de administrador |
| **Descripción** | Construye y levanta los contenedores necesarios para Petitoya utilizando docker-compose up -d. Verifica que los contenedores estén funcionando correctamente con el comando docker ps. |
| **Paso 6** | |
| **Tipo** | Configuración de la base de datos |
| **Componente** | Base de datos MySQL y Phpmyadmin |
| **Permisos** | Permisos de administrador |
| **Descripción** | abrir el cliente de phpmyadmin del contenedor de docker (usuario:root contraseña:12345). importando el archivo SQL proporcionado (petitoya.sql) se encontrara toda la base de datos lista para usarse, |

| **Paso 7** | |
| --- | --- |
| **Tipo** | ejecución de backend |
| **Componente** | funcionamiento de backend |
| **Permisos** | Permisos de administrador |
| **Descripción** | Después de configurar y ejecutar exitosamente MySQL, deberás reiniciar el contenedor del backend para asegurarte de que todos los cambios en la base de datos se integren correctamente. Utiliza el comando docker-compose restart backend para reiniciar el contenedor. Verifica que el backend esté funcionando correctamente comprobando los logs del contenedor y asegurándote de que no haya errores. |
| **Paso 8** | |
| **Tipo** | Verificación |
| **Componente** | Sistema completo |
| **Permisos** | Usuario administrador |
| **Descripción** | Verifica que el sistema Petitoya esté funcionando correctamente realizando pruebas básicas. Asegúrate de que los pedidos se puedan realizar, los datos se almacenan correctamente en la base de datos, y todas las funcionalidades principales están operativas. |

# VERIFICACIÓN DEL PROCESO DE INSTALACIÓN

La verificación del proceso de instalación asegura que el sistema Petitoya esté funcionando según lo esperado y que todos los componentes se hayan instalado y configurado correctamente. Este proceso implica realizar pruebas básicas y revisar la funcionalidad del sistema para identificar cualquier problema potencial.

| **Verificacion General** | |
| --- | --- |
| **Paso 1** | |
| **Tipo** | Verificación del contenedor |
| **Componente** | Contenedores Docker |
| **Permisos** | Usuario administrador |
| **Descripción** | Verifica que todos los contenedores necesarios para Petitoya estén en funcionamiento. Utiliza el comando docker ps para listar los contenedores activos. Asegúrate de que los contenedores del backend, frontend y la base de datos estén en estado "Up". |
| **Paso 2** | |
| **Tipo** | Revisión de logs |
| **Componente** | Logs de contenedores |
| **Permisos** | Usuario administrador |
| **Descripción** | Revisa los logs de los contenedores para identificar posibles errores o advertencias. Utiliza los comandos docker logs [nombre\_del\_contenedor] para acceder a los logs específicos de cada contenedor (por ejemplo, docker logs backend). |
| **Pruebas de Funcionalidad** | |
| **Paso 1** | |
| **Tipo** | Prueba de acceso a la interfaz |
| **Componente** | Interfaz de usuario de Petitoya |
| **Permisos** | Usuario administrador |
| **Descripción** | Accede a la interfaz de usuario de Petitoya a través del navegador web. Verifica que la página de inicio cargue correctamente y que todos los elementos de la interfaz estén visibles y funcionales. |
| **Paso 2** | |
| **Tipo** | Prueba de inicio de sesión |
| **Componente** | Funcionalidad de inicio de sesión |
| **Permisos** | Usuario administrador |
| **Descripción** | Realiza el inicio de sesión con las credenciales de administrador para confirmar que el sistema autentica correctamente a los usuarios. Asegúrate de que puedas acceder a todas las funciones administrativas. |
| **Paso 3** | |
| **Tipo** | Prueba de gestión de pedidos |
| **Componente** | Funcionalidad de gestión de pedidos |
| **Permisos** | Usuario administrador |
| **Descripción** | Crea un pedido en el sistema. Verifica que el pedido se guarde correctamente en la base de datos y que los datos se reflejen en la interfaz de usuario. |
| **Paso 4** | |
| **Tipo** | Prueba de generación de informes |
| **Componente** | Funcionalidad de informes |
| **Permisos** | Usuario administrador |
| **Descripción** | Genera un informe de ventas o de productos desde la interfaz de administración. Asegúrate de que el informe se genere correctamente y que los datos sean precisos. |
| **Verificación de la Base de Datos** | |
| **Paso 1** | |
| **Tipo** | Verificación de tablas |
| **Componente** | Base de datos MySQL |
| **Permisos** | Usuario administrador |
| **Descripción** | Accede a la base de datos MySQL y verifica que todas las tablas necesarias se hayan creado correctamente. Utiliza una herramienta de administración de base de datos como phpMyAdmin o la línea de comandos MySQL.(la informacion sobre la base de datos puedes encontrarlo en la carpeta “documentacion” del repositorio del software). |
| **Paso 2** | |
| **Tipo** | Verificación de datos |
| **Componente** | Datos en la base de datos |
| **Permisos** | Usuario administrador |
| **Descripción** | Revisa algunos registros en las tablas para asegurarte de que los datos se están almacenando correctamente. Verifica que los datos de prueba ingresados durante la instalación se reflejen adecuadamente. |

# MARCHA ATRÁS DE LA INSTALACIÓN Y CONFIGURACIÓN

La marcha atrás de la instalación y configuración es el proceso de deshacer los cambios realizados durante la instalación de Petitoya y restaurar el sistema a su estado anterior. Este procedimiento es crucial en caso de fallos durante la instalación, cambios incorrectos en la configuración, o cuando se necesita desinstalar el sistema por cualquier motivo.

## Requisitos previos

| **Requisito** | **Descripción** |
| --- | --- |
| **Acceso** | Acceso administrativo al servidor y a los sistemas relacionados. |
| **Copia de seguridad** | Tener una copia de seguridad reciente de los datos y la configuración del sistema. |
| **Documentación** | Documentación sobre la configuración actual y los pasos realizados durante la instalación. |

## Marcha atrás del sistema

< Cumplimentar tabla y eliminar cuadro>



| **Procedimiento de marcha atrás** | |
| --- | --- |
| **Paso 1** | |
| **Tipo** | Desinstalación de contenedores |
| **Componente** | Contenedores Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Detén y elimina los contenedores Docker que fueron creados para Petitoya. Utiliza el comando docker-compose down para detener y eliminar todos los contenedores asociados al proyecto. Asegúrate de que todos los contenedores estén eliminados revisando con docker ps -a. |

| **Paso 2** | |
| --- | --- |
| **Tipo** | Eliminación de imágenes de Docker |
| **Componente** | Imágenes de Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Elimina las imágenes de Docker asociadas con Petitoya utilizando el comando docker rmi [nombre\_imagen]. Revisa las imágenes existentes con docker images para identificar las imágenes que deben eliminarse. |
| **Paso 3** | |
| **Tipo** | Eliminación de archivos del proyecto |
| **Componente** | Archivos del proyecto Petitoya |
| **Permisos** | Permisos de administrador |
| **Descripción** | Elimina los archivos del proyecto Petitoya del servidor. Navega a la carpeta donde se clonó el repositorio y elimina todos los archivos y carpetas asociados con el comando rm -rf [nombre\_carpeta].(en caso de usar linux) para windows puedes usar el explorador de archivos, vas a la ubicacion donde clonaste el repositorio y borras la carpeta. |
| **Paso 4** | |
| **Tipo** | Restauración de la base de datos |
| **Componente** | Base de datos MySQL |
| **Permisos** | Permisos de administrador |
| **Descripción** | Si se realizaron cambios en la base de datos durante la instalación, restaura la base de datos desde la copia de seguridad más reciente. Utiliza las herramientas de administración de MySQL para importar la copia de seguridad. |

## Marcha atrás del software base

| **Paso 1** | |
| --- | --- |
| **Tipo** | Eliminación de contenedores |
| **Componente** | Contenedores Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Detén y elimina los contenedores Docker creados para Petitoya. Usa el comando docker-compose down para detener y eliminar todos los contenedores asociados al proyecto. |
| **Paso 2** | |
| **Tipo** | Eliminación de volúmenes de Docker |
| **Componente** | Volúmenes Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Elimina los volúmenes de Docker para una limpieza completa,Usa docker volume rm mysql\_data y lista los volúmenes existentes con docker volume ls, para verificar que se haya eliminado el volumen correctamente. |
| **Paso 3** | |
| **Tipo** | Reconstrucción de contenedores |
| **Componente** | Contenedores Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Reconstruye los contenedores desde cero. Navega al directorio del proyecto y ejecuta docker-compose up -d para iniciar los contenedores con la configuración predeterminada. |
| **Paso 4** | |
| **Tipo** | Verificación del entorno |
| **Componente** | Entorno Docker |
| **Permisos** | Permisos de administrador |
| **Descripción** | Verifica que todos los contenedores estén funcionando correctamente después de la reconstrucción. Usa docker ps para listar los contenedores activos y revisa los logs con docker logs [nombre\_contenedor] para asegurarte de que no haya errores. |
| **Paso 5** | |
| **Tipo** | Restauración de datos |
| **Componente** | Base de datos MySQL |
| **Permisos** | Permisos de administrador |
| **Descripción** | Si se realizaron cambios en la base de datos y necesitas restaurar datos previos, importa la copia de seguridad más reciente en el contenedor de MySQL. Utiliza las herramientas de administración de MySQL para este proceso. |

# ANEXOS

### Resumen de Tareas de Configuración

**Preparación del Entorno**

* **Acción**: Verificar que el servidor cumpla con los requisitos de hardware y software.
* **Requisitos**: Asegurarse de que Docker y Docker Compose estén instalados.

**Configuración de Docker**

* **Acción**: Descargar el código fuente de Petitoya.
* **Requisitos**: Tener Docker y Docker Compose instalados.
* **Instrucción**: Ejecutar docker-compose build para construir las imágenes necesarias.

**Configuración de la Base de Datos**

* **Acción**: Importar el archivo petitoya.sql en el contenedor de MySQL.
* **Requisitos**: Acceso a la base de datos y a herramientas de administración MySQL.
* **Instrucción**: Configurar las credenciales en el archivo de configuración.

**Configuración del Sistema**

* **Acción**: Acceder a la interfaz de administración de Petitoya.
* **Instrucción**: Configurar los parámetros del sistema y personalizar los productos disponibles.

**Verificación**

* **Acción**: Verificar el estado de los contenedores.
* **Instrucción**: Usar docker ps para listar contenedores activos y revisar los logs para asegurar un funcionamiento correcto.

### Archivos y Scripts Adicionales

* **docker-compose.yml**:
  + **Descripción**: Archivo de configuración para Docker Compose que define los servicios, redes y volúmenes necesarios para Petitoya.
* **Dockerfile**:
  + **Descripción**: Script utilizado para construir las imágenes de Docker para el sistema Petitoya.
* **petitoya.sql**:
  + **Descripción**: Script SQL para la inicialización de la base de datos MySQL, incluyendo la creación de tablas y carga de datos.

## Resumen de tareas de configuración

### Preparación del Entorno

* **Verificación de Requisitos de Hardware y Software**:
  + **Descripción**: Asegúrate de que el servidor cumpla con los requisitos mínimos y recomendados especificados en la sección de recursos hardware y software del manual.
  + **Acciones**:
    - Revisar especificaciones del servidor.
    - Instalar Docker y Docker Compose en el servidor si aún no están instalados.

### Configuración de Docker

* **Descarga del Código Fuente**:
  + **Descripción**: Obtén el código fuente de Petitoya desde el repositorio oficial.
  + **Acciones**:
    - Clonar el repositorio o descargar el archivo comprimido.
* **Construcción de Imágenes Docker**:
  + **Descripción**: Prepara el entorno Docker construyendo las imágenes necesarias.
  + **Acciones**:
    - Ejecutar el comando docker-compose build en el directorio del proyecto para construir las imágenes.

### Configuración de la Base de Datos

* **Importación del Archivo SQL**:
  + **Descripción**: Inicializa la base de datos MySQL importando el archivo de inicialización.
  + **Acciones**:
    - Importar el archivo petitoya.sql en el contenedor de MySQL utilizando herramientas de administración de bases de datos.
    - Asegurarse de que las tablas y datos se hayan creado correctamente.
* **Configuración de Credenciales**:
  + **Descripción**: Configura las credenciales de acceso a la base de datos en el archivo de configuración.
  + **Acciones**:
    - Editar el archivo de configuración para agregar las credenciales correctas para la conexión a MySQL.

### 

### 

### 

### 

### Configuración del Sistema

* **Acceso a la Interfaz de Administración**:
  + **Descripción**: Accede a la interfaz de administración de Petitoya para realizar configuraciones iniciales.
  + **Acciones**:
    - Iniciar sesión en la interfaz web de Petitoya.
* **Configuración de Parámetros del Sistema**:
  + **Descripción**: Ajusta los parámetros del sistema y personaliza los productos disponibles.
  + **Acciones**:
    - Configurar ajustes generales del sistema (por ejemplo, puerto de conexión).
    - Personalizar los productos y opciones disponibles según las necesidades del negocio.

### Verificación

* **Verificación del Estado de Contenedores**:
  + **Descripción**: Asegúrate de que todos los contenedores necesarios estén en funcionamiento.
  + **Acciones**:
    - Ejecutar docker ps para listar los contenedores activos.
    - Verificar que todos los servicios estén funcionando correctamente.
* **Revisión de Logs**:
  + **Descripción**: Revisa los logs de los contenedores para identificar posibles errores o problemas.
  + **Acciones**:
    - Usar docker logs [nombre del contenedor] para ver los logs y solucionar problemas según sea necesario.

# GLOSARIO

<Insertar comentario y eliminar cuadro>

| **Término** | **Descripción** |
| --- | --- |
| API | Interfaz de Programación de Aplicaciones. Conjunto de definiciones y protocolos que permite la comunicación entre diferentes sistemas o aplicaciones. |
| contenedor | Entorno ligero y aislado en el que se ejecutan aplicaciones, asegurando que funcionen de manera consistente sin importar el entorno. Utiliza Docker para la virtualización. |
| Docker | Plataforma para desarrollar, enviar y ejecutar aplicaciones dentro de contenedores. Facilita la creación y gestión de entornos consistentes para el software. |
| Docker compose | Herramienta que permite definir y ejecutar aplicaciones Docker multicontenedor mediante un archivo de configuración YAML. |
| base de datos | Sistema que almacena, organiza y gestiona datos de manera estructurada. En este manual, se utiliza MySQL como base de datos para Petitoya. |
| imagen docker | Archivo que contiene todo el entorno necesario para ejecutar una aplicación, incluyendo el código, dependencias y configuraciones. Se utiliza para crear contenedores. |
| servidor web | Software que aloja y sirve aplicaciones web. En este caso, se utiliza Apache para gestionar las solicitudes HTTP de los usuarios y servir la aplicación Petitoya. |
| SQL | Lenguaje de Consulta Estructurado. Utilizado para gestionar y manipular bases de datos relacionales. El archivo petitoya.sql se utiliza para inicializar la base de datos. |
| Vite | Herramienta de construcción y desarrollo de frontend que acelera el proceso de desarrollo con una rápida recarga en caliente y optimización del código. |
| MERN | Stack tecnológico que combina MongoDB, Express.js, React y Node.js para el desarrollo de aplicaciones web completas. Aunque en este caso se utiliza MySQL en lugar de MongoDB. |
| Backend | parte del software que maneja la lógica de la aplicación, interacción con bases de datos y procesamiento de datos, generalmente ejecutado en el servidor. |
| Frontend | Parte del software que interactúa directamente con el usuario, proporcionando la interfaz gráfica y la experiencia de usuario. |
| Configuracion | Proceso de ajuste de parámetros y ajustes en software y hardware para asegurar que funcionen correctamente según las especificaciones |
| Contenedor docker | Unidad de software que empaqueta código y todas sus dependencias, asegurando que la aplicación funcione de manera consistente sin importar el entorno. |
| Permisos de administrador | Privilegios necesarios para realizar cambios importantes en el sistema, como la instalación de software, configuración de servicios y gestión de datos. |
| repositorio | Almacén centralizado para guardar y gestionar el código fuente y otros recursos relacionados con el desarrollo de software. |
| Script Sql | Archivo que contiene instrucciones en SQL para crear, modificar y administrar bases de datos y sus contenidos. |

# BIBLIOGRAFÍA Y REFERENCIAS

<Insertar comentario y eliminar cuadro>



| **Referencia** | **Título** | **Código** |
| --- | --- | --- |
| Ref. 1 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |