

Instalación con Docker

Instalar Drupal Commerce con Docker

Anonymous · 25/09/2026

Instalar Drupal Commerce con Docker

Sigue la [guía de los stacks de Docker](#) para instalar Drupal Commerce con nuestro [stack](#) y ten en cuenta lo propio de esta plataforma. El detalle completo está en el README del repositorio.

Recursos

El stack necesita unos 512 MB de RAM y 700 MB de disco para las imágenes. En reposo usa unos 250 MiB en total (MariaDB ~160 MiB, PHP-FPM ~70 MiB, Caddy ~15 MiB y cron ~5 MiB).

La imagen de Drupal Commerce se construye en tu servidor, por lo que el stack se levanta con `docker compose up -d --build` en vez de `docker compose up -d`. La primera construcción tarda aproximadamente un minuto y genera una imagen de unos 240 MB. La instalación inicial necesita acceso a internet para descargar las traducciones al español desde localize.drupal.org.

Variables obligatorias

Variable	Qué debe contener
DRUPAL_URL	URL pública de la tienda, por ejemplo <code>https://tienda.example.com</code> .
SITE_ADDRESS	El dominio de la tienda, por ejemplo <code>tienda.example.com</code> .
DRUPAL_HASH_SALT	Clave secreta de Drupal. Si la cambias, se cierran todas las sesiones. No se incluye en los respaldos: guarda una copia.
DB_PASSWORD y DB_ROOT_PASSWORD	Contraseñas de la base de datos.
DRUPAL_ADMIN_EMAIL y DRUPAL_ADMIN_PASSWORD	Correo y contraseña del administrador, que se crea en la instalación inicial. La contraseña solo se aplica entonces.
SMTP_HOST	Recomendada: el stack arranca sin ella, pero la plataforma no podrá enviar correos. Servidor SMTP. Completa también <code>SMTP_PORT</code> , <code>SMTP_SECURE</code> , <code>SMTP_USER</code> , <code>SMTP_PASSWORD</code> y <code>SMTP_FROM</code> según tu proveedor. <code>SMTP_FROM</code> (por defecto, <code>DRUPAL_ADMIN_EMAIL</code>) es el correo del sitio en cada petición, pero el de la tienda solo se define al crearla; después se cambia desde el panel.

Drupal solo responde al dominio de `DRUPAL_URL`. Si la tienda también se usa con otro nombre (por ejemplo, `www`), agrégalo en `DRUPAL_EXTRA_HOSTS`.

Configuración inicial de la tienda

La tienda se crea configurada para Chile: idioma `es`, moneda `CLP`, país `CL` y ciudad Santiago, IVA de 19 % incluido en los precios y el medio de pago manual “Transferencia bancaria”. Si vendes en otro país, ajusta antes del primer arranque `DRUPAL_LOCALE`, `COMMERCE_CURRENCY`, `COMMERCE_COUNTRY`, `COMMERCE_CITY`, las variables `COMMERCE_TAX_*` y `COMMERCE_PRICES_INCLUDE_TAX`. Cada elemento se crea solo si no existe, y después se modifica desde el panel.

`setup` revisa la moneda y el impuesto en cada arranque: si cambias `COMMERCE_CURRENCY` se importa otra moneda, y si cambias `COMMERCE_TAX_NAME` se crea un segundo impuesto. Después de la instalación, modifica la tienda y el impuesto desde el panel, no en el `.env`. `TIMEZONE` se aplica a PHP en cada arranque, pero la zona horaria del sitio y de la tienda solo se define al crear la tienda.

Acceso

Cuando `setup` termina, la tienda queda disponible en tu dominio. Inicia sesión en `/user/login` con el usuario `admin` (configurable con `DRUPAL_ADMIN_USER`) y administra la tienda en `/admin/commerce`.

Actualizaciones

Drupal y Commerce se actualizan cambiando la versión en el `.env`. Respalda primero, cambia `DRUPAL_VERSION` o `COMMERCE_VERSION` y ejecuta `docker compose up -d --build`: la imagen se reconstruye y `setup` ejecuta `drush updatedb` antes de que arranque PHP-FPM. Revisa las notas de cada versión menor de Drupal y de Commerce.

Los módulos y temas tampoco se instalan desde el panel: agrégalos como paquetes de Composer en `DRUPAL_EXTRA_PACKAGES` (separados por espacios), reconstruye con `docker compose up -d --build` y actívalos con Drush, por ejemplo:

```
docker compose exec -u www-data php drush pm:install commerce_shipping
```

Reconstruye la imagen periódicamente con `docker compose build --pull` para recibir las correcciones de seguridad de PHP y Alpine, y sube las versiones cuando haya actualizaciones de seguridad de Drupal.

Restaurar un respaldo

Los respaldos incluyen la base de datos y los archivos públicos y privados. `DRUPAL_HASH_SALT` no se incluye: conserva tu `.env`. Para restaurar uno:

```
docker compose stop php cron
docker compose run --rm --no-deps backup restore <fecha>
docker compose up -d
docker compose exec -u www-data php drush cache:rebuild
```

La base de datos debe estar en ejecución (`docker compose up -d db` si el stack está detenido). La restauración borra primero todas las tablas, así que no queda nada de lo creado después del respaldo.

Notas

- Los envíos no son parte de Commerce: para usarlos, agrega `drupal/commerce_shipping` a `DRUPAL_EXTRA_PACKAGES`, actívalo y configura los métodos de envío en el panel.
- La traducción al español de Drupal es de España (“Añadir a la cesta”, “Tramitar compra”). Puedes editar los textos en `/admin/config/regional/translate`.

Siguiente paso

Con Drupal Commerce funcionando, configúralo como origen de datos en BillMySales siguiendo la [configuración inicial](#).

Última actualización el 25/09/2026