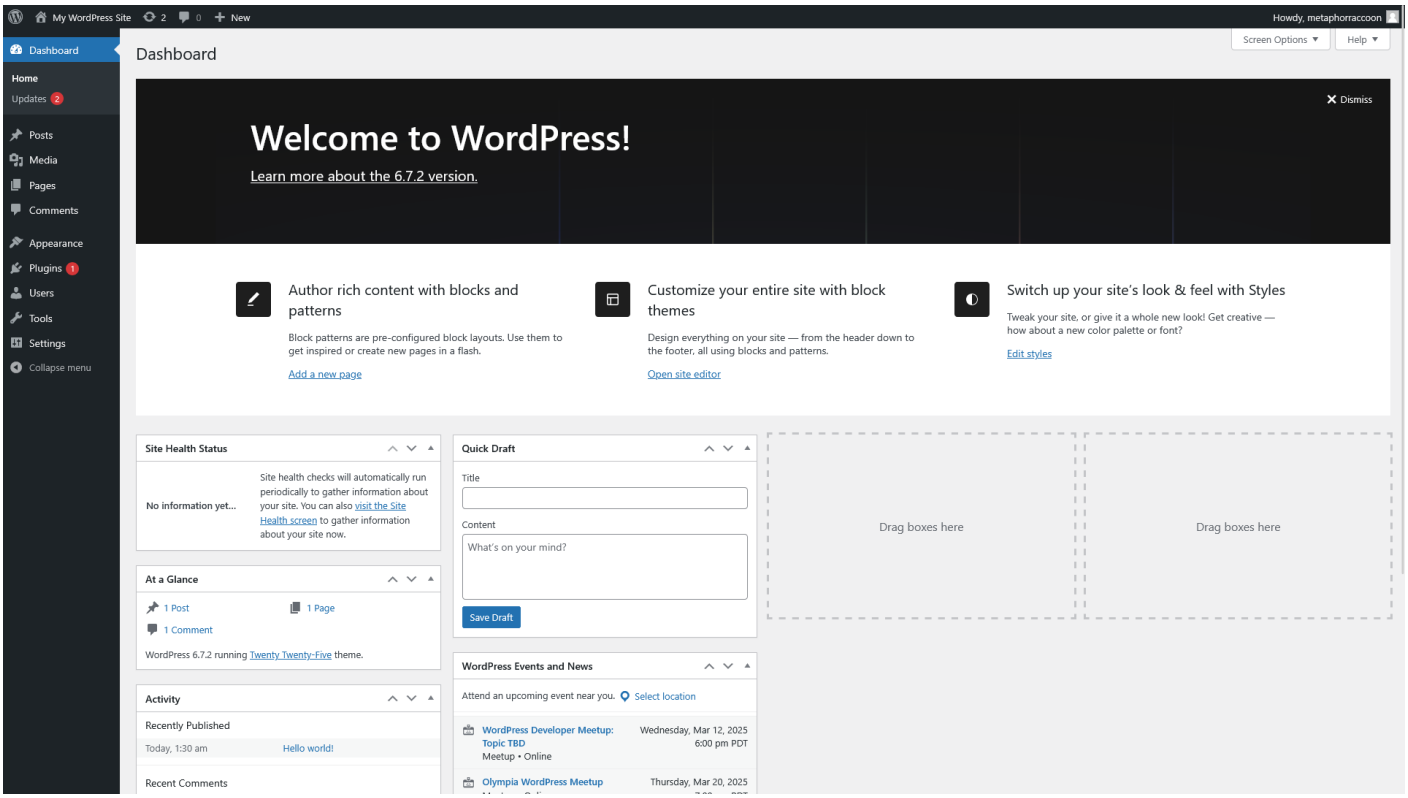
WordPress •

WordPress is an open-source publishing platform that allows you to easily build a website and blog. Leverage a range of pre-made themes and visual block system to build your site.

Overview



WordPress is an open-source publishing platform that allows you to easily build a website and blog. Leverage a range of pre-made themes and visual block system to build your site.

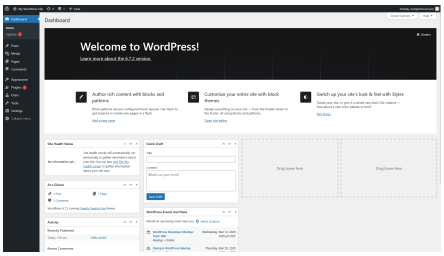
> Tryout an WordPress demo.

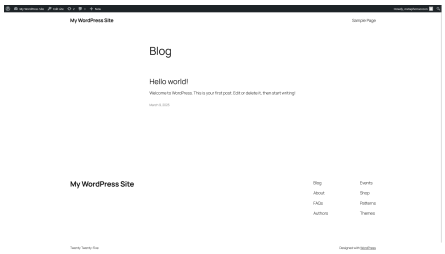
- Preview a website in real-time as you're editing to and schedule changes in the future.
- Browse over 50,000 add-ons for your WordPress website.
- Start from a pre-made Pattern and edit it to better fit your needs.

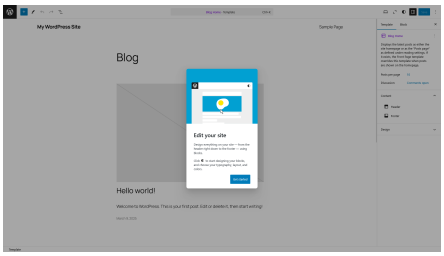
Media

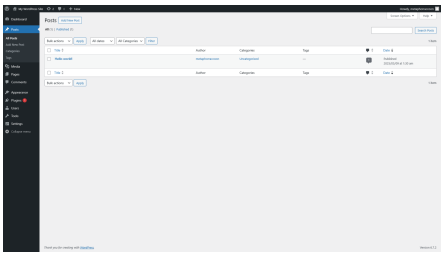
Screenshots

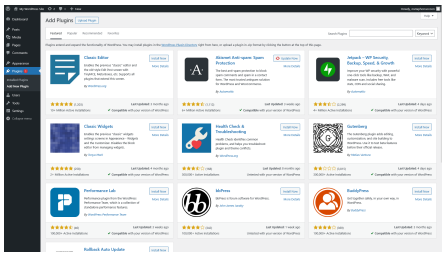
> WordPress v6.7.2

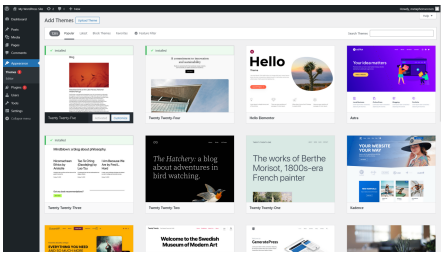












Setup & Configuration

We need to install the service through Portainer and configure any necessary settings.

Preparation

There are some things we need to do in preparation to install this service.

Volumes

Persistent Data

This is where the service will store its own application data and ensures we can quickly update the service image.

> Ensure your user has permissions to access the folder.

Environment

DB_USER

This is the username to be used for logging into the database.

> *For example:* wordpress_admin

Passwords

> Keep these securely stored in a password manager, such as VaultWarden.

DB_PASSWORD

This is the password that will be used for access to the database.

> It is important to use secure, randomly generated password.

You can use a random alphanumeric string from a password manager, or open the terminal and run the command:

```
tr -dc 'A-Za-z0-9!"#$%&'\''()*+,-./:;<=>?@[\]^_`{|}~' </dev/urandom | head -c 32; echo
```

This pulls a random string from the 'urandom' device, removes unwanted characters and trim it to an appropriate length.

Installation

The service can be installed through the Portainer web interface.

> Learn about creating a new stack.

Docker Compose

Use the following code to install the service:

```
---
services:
  db:
    image: mariadb:10.6.4-focal
    command: '--default-authentication-plugin=mysql_native_password'
    volumes:
      - /srv/wordpress/db:/var/lib/mysql
    restart: always
    environment:
      - MYSQL_ROOT_PASSWORD=
      - MYSQL_DATABASE=wordpress
      - MYSQL_USER=${DB_USER}
      - MYSQL_PASSWORD=${DB_PASSWORD}
    expose:
      - 3306

  wordpress:
    image: wordpress:latest
    ports:
      - 8080:80
    restart: always
    depends_on:
      - db
```

```
    volumes:
     - /srv/wordpress/html:/var/www/html
    environment:
      - WORDPRESS_DB_HOST=db
      - WORDPRESS_DB_USER=${DB_USER}
      - WORDPRESS_DB_PASSWORD=${DB_PASSWORD}
      - WORDPRESS_DB_NAME=wordpress
```

Environment

Use the following environment to configure the service using the values prepared earlier:

```
# Database Configuration
DB_USER=wordpress_admin

# Passwords
# After the initial setup, the passwords should be saved elsewhere and stack should be re-deployed with these settings deleted.

# Database Passwords
DB_PASSWORD=
```

Updating

> Re-Deploy the Stack

This service has been optimized for running in Docker.

This allows you to re-deploy the stack through Portainer to download the latest updates.

User Manual

Development

> This software is released under the MIT license.

You can learn more about how to contribute to WordPress through their website.

The WordPress Foundation also accepts donations.

Resources

Official

- Official Documentation
- Official Forum
- Official GitHub Repository
- Official Website
- Wikipedia Entry
- Installation Guide
- Frequently Asked Questions