

Name: - Harshal Pawar

#829087 - Project: Install MySQL on OpenSUSE and WordPress on Ubuntu with Database Connectivity

Description:

A project requiring the installation of MySQL on an OpenSUSE system and WordPress on an Ubuntu system. The goal is to establish connectivity between the two systems, allowing WordPress on Ubuntu to use MySQL as its database server running on OpenSUSE. A detailed documentation should be created to explain the entire process, including configurations and troubleshooting steps.

Requirements:

- Install MySQL on the OpenSUSE system and configure it for remote access.
- Install WordPress on the Ubuntu system and configure it to connect to the MySQL database on the OpenSUSE system.
- Ensure proper firewall and network settings to enable communication between the two systems.
- Verify connectivity by confirming that WordPress can read and write data to the MySQL database.
- Create detailed documentation covering the installation process for both systems, database configuration, connectivity setup, and troubleshooting advice.

Use Case:

This setup will allow WordPress, running on an Ubuntu server, to use MySQL for database management hosted on an OpenSUSE system, ideal for distributed web application environments.

Part 1: Install and Configure MySQL on OpenSUSE

Update the OpenSUSE repositories to ensure the latest packages are installed.

```

opensuse:~ # zypper refresh
Repository 'openSUSE-Leap-15.0-Non-Oss' is up to date.
Repository 'openSUSE-Leap-15.0-Oss' is up to date.
Repository 'openSUSE-Leap-15.0-Update' is up to date.
Repository 'openSUSE-Leap-15.0-Update-Non-Oss' is up to date.
All repositories have been refreshed.
opensuse:~ # zypper update
Loading repository data...
Warning: Repository 'openSUSE-Leap-15.0-Update' appears to be outdated. Consider using a different mirror or server.
Warning: Repository 'openSUSE-Leap-15.0-Update-Non-Oss' appears to be outdated. Consider using a different mirror or server.
Reading installed packages...

Nothing to do.

```

Install Mariadb.

```

opensuse:~ # zypper install mariadb mariadb-client mariadb-server
Loading repository data...
Warning: Repository 'openSUSE-Leap-15.0-Update' appears to be outdated. Consider using a different mirror or server.
Warning: Repository 'openSUSE-Leap-15.0-Update-Non-Oss' appears to be outdated. Consider using a different mirror or server.
Reading installed packages...
'mariadb-server' not found in package names. Trying capabilities.
Resolving package dependencies...

The following 4 NEW packages are going to be installed:
  libmariadb3 mariadb mariadb-client mariadb-errormessages

4 new packages to install.
Overall download size: 19.1 MiB. Already cached: 0 B. After the operation, additional 150.7 MiB will be used.
Continue? [y/n/v/...? shows all options] (y): y
Retrieving package libmariadb3-3.1.2-lp150.10.1.x86_64                      (1/4), 127.2 KiB (3
Retrieving: libmariadb3-3.1.2-lp150.10.1.x86_64.rpm .....[
Retrieving package mariadb-errormessages-10.2.25-lp150.2.16.1.noarch      (2/4), 212.7 KiB (
Retrieving: mariadb-errormessages-10.2.25-lp150.2.16.1.noarch.rpm .....[d

```

Ensure MariaDB is installed and running

```

opensuse:~ # systemctl start mariadb
opensuse:~ # systemctl enable mariadb
opensuse:~ # systemctl status mariadb
● mariadb.service - MySQL server
   Loaded: loaded (/usr/lib/systemd/system/mariadb.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2024-12-12 10:26:56 IST; 7s ago
     Main PID: 9964 (mysqld)
    Status: "Taking your SQL requests now..."
       Tasks: 30 (limit: 4915)
      CGroup: /system.slice/mariadb.service
              └─9964 /usr/sbin/mysqld --defaults-file=/etc/my.cnf --user=mysql

Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: rcmysql start
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: You can test the MariaDB daemon
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: Please report any problems at
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: The latest information about M
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: You can find additional inform
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: http://dev.mysql.com
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: Consider joining MariaDB's str
Dec 12 10:26:49 opensuse.example.com mysql-systemd-helper[9885]: https://mariadb.org/get-involv
Dec 12 10:26:56 opensuse.example.com mysql-systemd-helper[9964]: 2024-12-12 10:26:56 1397504871
Dec 12 10:26:56 opensuse.example.com systemd[1]: Started MySQL server.
opensuse:~ #

```

Secure MariaDB Installation

```
opensuse:~ # mysql_secure_installation
```

```
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

```
In order to log into MariaDB to secure it, we'll need the current  
password for the root user. If you've just installed MariaDB, and  
you haven't set the root password yet, the password will be blank,  
so you should just press enter here.
```

```
Enter current password for root (enter for none):  
OK, successfully used password, moving on...
```

```
Setting the root password ensures that nobody can log into the MariaDB  
root user without the proper authorisation.
```

- **Set a root password:** Set a secure password for the root user.
- **Remove anonymous users:** Prevent anonymous logins.
- **Disallow remote root login:** For security, remote root login is disabled.
- **Remove test databases:** Delete unnecessary test databases.

Log in to MariaDB using the root user and create a Database and User for WordPress.

```
opensuse:~ # mysql -u root -p
```

```
Enter password:
```

```
Welcome to the MariaDB monitor. Commands end with ; or \g.
```

```
Your MariaDB connection id is 17
```

```
Server version: 10.2.25-MariaDB SUSE package
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [(none)]> show databases;
```

```
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
+-----+  
3 rows in set (0.01 sec)
```

Run the following SQL commands to create a database and a user with privileges.

CREATE DATABASE wordpress_db; -- Creates the WordPress database.

CREATE USER 'user2'@'%' IDENTIFIED BY '12345'; -- Creates a user with remote access.

GRANT ALL PRIVILEGES ON wordpress_db.* TO 'user2'@'%'; -- Grants privileges to the user.

FLUSH PRIVILEGES; -- Refreshes the privileges to apply changes

```
MariaDB [(none)]> CREATE DATABASE wordpress_db;  
Query OK, 1 row affected (0.00 sec)
```

```
MariaDB [(none)]> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| wordpress_db |  
+-----+  
4 rows in set (0.01 sec)
```

```
MariaDB [(none)]> CREATE USER 'user2'@'%' IDENTIFIED BY '12345';  
Query OK, 0 rows affected (0.01 sec)
```

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON wordpress_db.* TO 'user2'@'%;  
Query OK, 0 rows affected (0.00 sec)
```

```
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.00 sec)
```

Edit the MariaDB configuration file.

Look for the bind-address directive and change it to the server's IP or 0.0.0.0:

```
opensuse:~ # cat /etc/my.cnf  
# The following options will be passed to all MariaDB clients  
[client]  
# Please note that storing the password in this file is not safe. For this  
# purpose you can, for example, list your password in the [client] section  
# of the '~/.my.cnf' configuration file with an access mode set to 400 or 600.  
# password      = your_password  
# port          = 3306  
# socket        = /run/mysql/mysql.sock  
  
# The MariaDB server  
[mysqld]  
  
# For security reasons, bind to 127.0.0.1 by default to enable networking  
# only on the loopback interface.  
bind-address    = 0.0.0.0
```

Restart MariaDB to apply changes

systemctl restart mariadb

Part 2: Install WordPress on Ubuntu

```
root@ubuntu:~# apt update
Hit:1 https://download.docker.com/linux/ubuntu focal InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [128 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [128 kB]
Get:5 http://in.archive.ubuntu.com/ubuntu focal-security InRelease [128 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages
```

Install Apache, PHP, and Required Modules

```
apt install apache2 php libapache2-mod-php php-mysql -y
```

Enable and start the Apache2 service.

```
sudo systemctl start apache2
```

```
sudo systemctl enable apache2
```

Install WordPress

```
root@ubuntu:~# wget https://wordpress.org/latest.tar.gz
--2024-12-12 05:44:38-- https://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 26931653 (26M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz          100%[=====] 25.68M  501KB/s
72s

2024-12-12 05:45:53 (367 KB/s) - 'latest.tar.gz' saved [26931653/26931653]
```

Extract WordPress file.

```
root@ubuntu:~# tar -xzf latest.tar.gz
wordpress/
wordpress/xmlrpc.php
wordpress/wp-blog-header.php
wordpress/readme.html
wordpress/wp-signup.php
wordpress/index.php
wordpress/wp-cron.php
wordpress/wp-config-sample.php
wordpress/wp-login.php
wordpress/wp-settings.php
wordpress/license.txt
wordpress/wp-content/
wordpress/wp-content/themes/
wordpress/wp-content/themes/twentytwentythree/
wordpress/wp-content/themes/twentytwentythree/theme.json
wordpress/wp-content/themes/twentytwentythree/parts/
wordpress/wp-content/themes/twentytwentythree/parts/footer.html
wordpress/wp-content/themes/twentytwentythree/parts/comments.html
wordpress/wp-content/themes/twentytwentythree/parts/header.html
wordpress/wp-content/themes/twentytwentythree/parts/post-meta.html
```

Set the correct permissions:

```
root@ubuntu:~# mv wordpress /var/www/html/  
root@ubuntu:~# chown -R www-data:www-data /var/www/html/wordpress  
root@ubuntu:~# chmod -R 755 /var/www/html/wordpress  
root@ubuntu:~# |
```

Configure Apache and create a new Apache configuration file.

Add the following content

```
root@ubuntu:~# cat /etc/apache2/sites-available/wordpress.conf  
<VirtualHost *:80>  
    ServerName 192.168.83.83  
    DocumentRoot /var/www/html/wordpress  
</VirtualHost>  
root@ubuntu:~# |
```

Configure WordPress

```
wp-config.php wp-config-sample.php  
root@ubuntu:/var/www/html/wordpress# cp wp-config-sample.php wp-config.php
```

Edit the configuration file and modify the following lines to connect to the MariaDB database

```
define('DB_NAME', 'wordpress_db');  
  
define('DB_USER', 'user2');  
  
define('DB_PASSWORD', '12345');  
  
define('DB_HOST', 'opensuse-server-ip');
```

```
root@ubuntu:/var/www/html/wordpress# cat wp-config.php  
<?php  
/**  
 * The base configuration for WordPress  
 *  
 */
```

```

// Database settings - You can get this info from your web host.
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress_db' );

/** Database username */
define( 'DB_USER', 'user2' );

/** Database password */
define( 'DB_PASSWORD', '12345' );

/** Database hostname */
define( 'DB_HOST', '192.168.83.15' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

/**#@+
 * Authentication unique keys and salts.

```

Enable the configuration and required modules

```

root@ubuntu:~# a2ensite wordpress
Enabling site wordpress.
To activate the new configuration, you need to run:
    systemctl reload apache2
root@ubuntu:~# systemctl reload apache2
root@ubuntu:~# a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
    systemctl restart apache2
root@ubuntu:~# systemctl restart apache2
root@ubuntu:~#

```

Part 3: Verify Connectivity and Access WordPress

Login the mysql on Ubuntu.

```
mysql -u user2 -p -h 192.168.83.15
```

Run the following commands:

```
SHOW DATABASES;
```

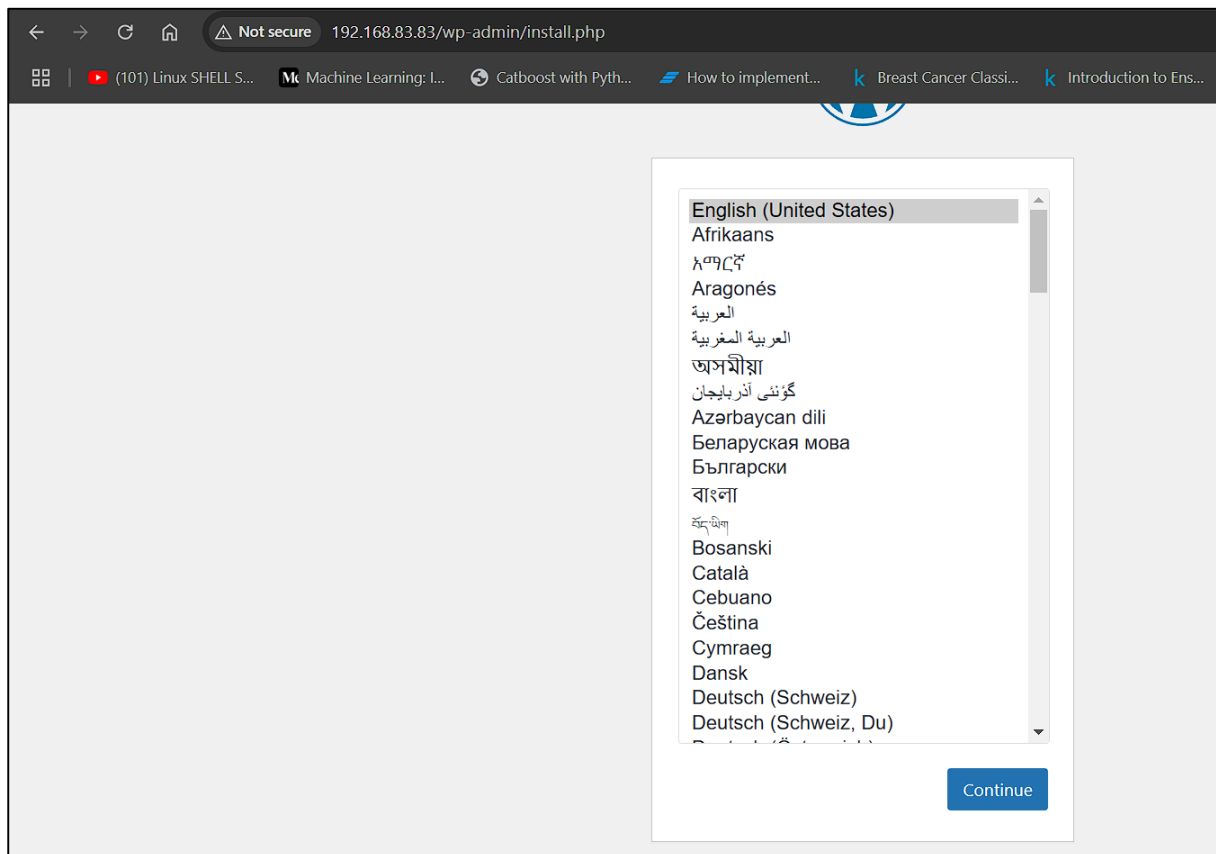
```
USE wordpress_db;
```

```
SELECT * FROM wp_users;
```

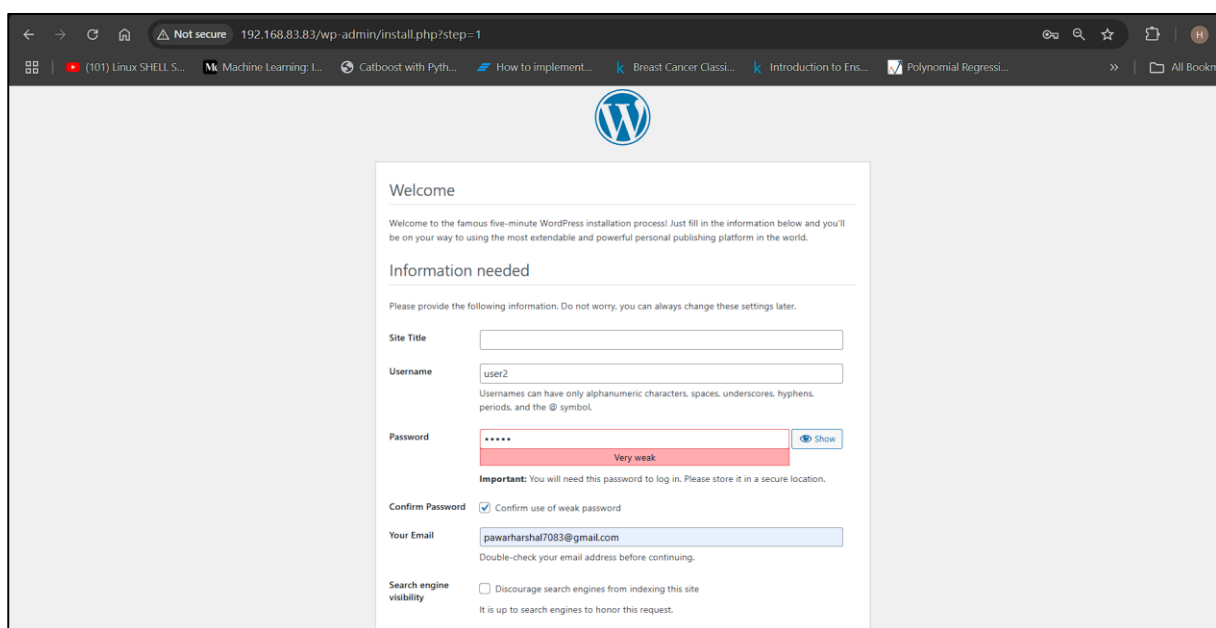

Open a web browser and navigate to <http://ubuntu-server-ip>

Follow the WordPress installation wizard.

Select the language as English.



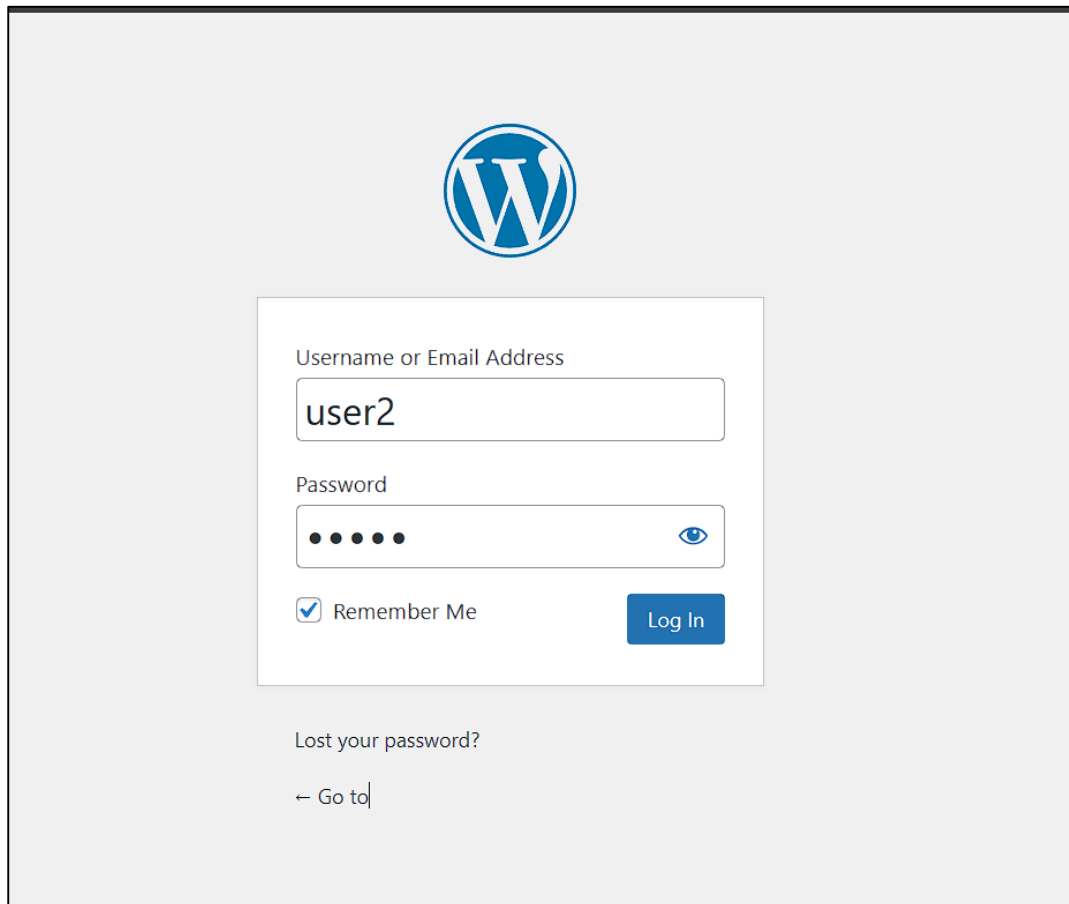
Set an admin username and password.



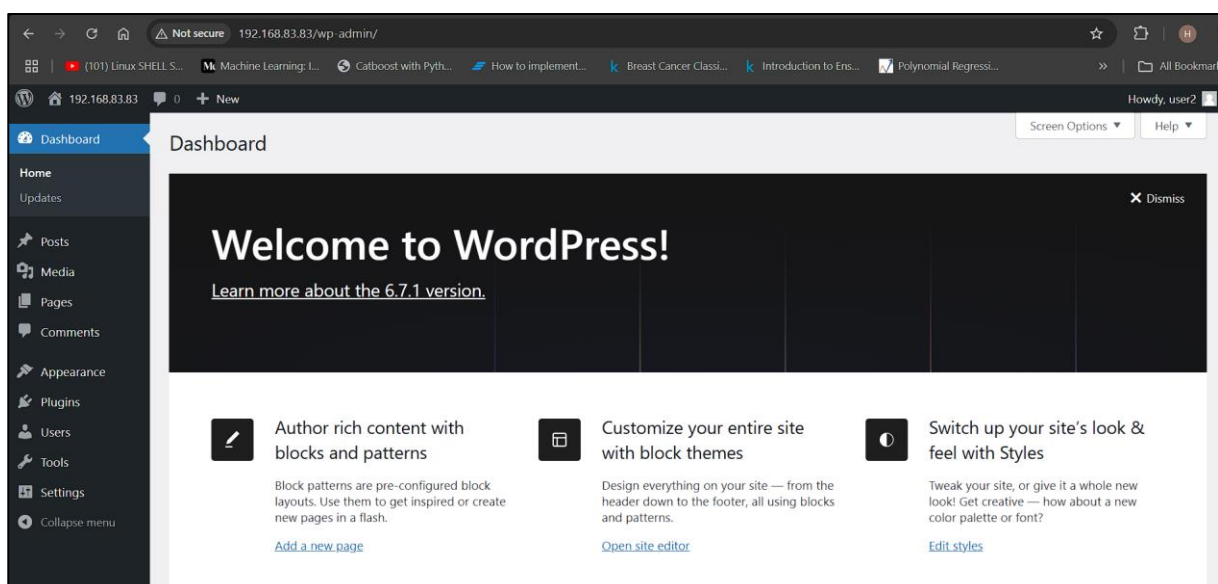
Put username and password and login and complete the setup.

Verify Data Interaction

Log in to the WordPress admin dashboard.

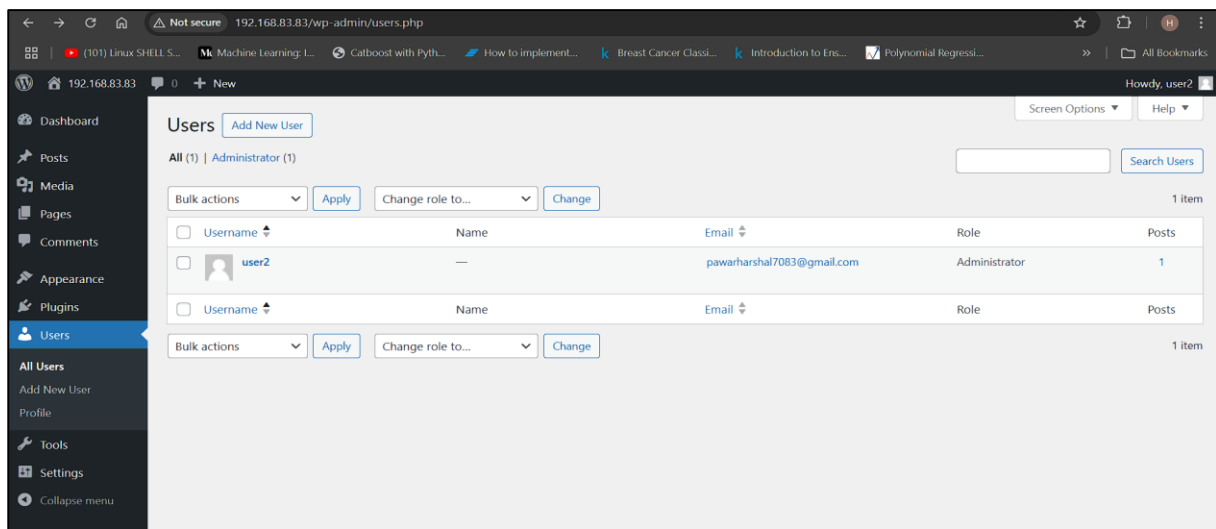


The image shows the WordPress login page. At the top center is the WordPress logo. Below it is a white box containing the login form. The form has two input fields: 'Username or Email Address' with the text 'user2' entered, and 'Password' with five dots representing the password. Below the password field is a checkbox labeled 'Remember Me' which is checked, and a blue 'Log In' button. Below the login box, there is a link 'Lost your password?' and a link '← Go to'.

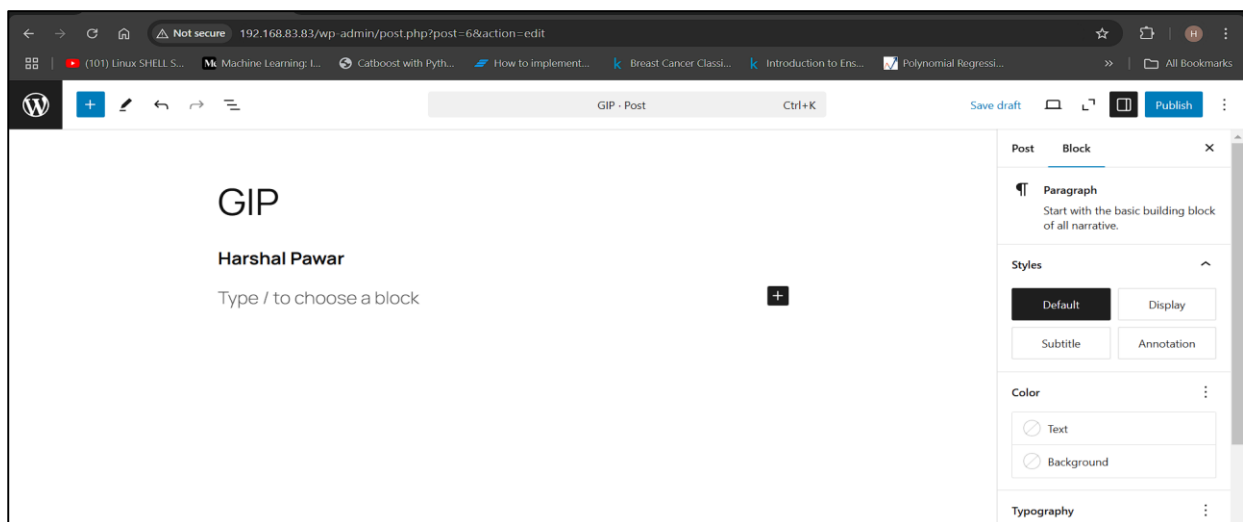


The image shows the WordPress admin dashboard. The browser address bar shows '192.168.83.83/wp-admin/'. The dashboard has a dark sidebar on the left with a menu: Dashboard, Home, Updates, Posts, Media, Pages, Comments, Appearance, Plugins, Users, Tools, Settings, and Collapse menu. The main content area has a 'Dashboard' title and a 'Welcome to WordPress!' message with a link to 'Learn more about the 6.7.1 version.'. Below the welcome message are three cards: 'Author rich content with blocks and patterns' with a link 'Add a new page', 'Customize your entire site with block themes' with a link 'Open site editor', and 'Switch up your site's look & feel with Styles' with a link 'Edit styles'. The top right of the dashboard shows 'Howdy, user2' and links for 'Screen Options' and 'Help'.

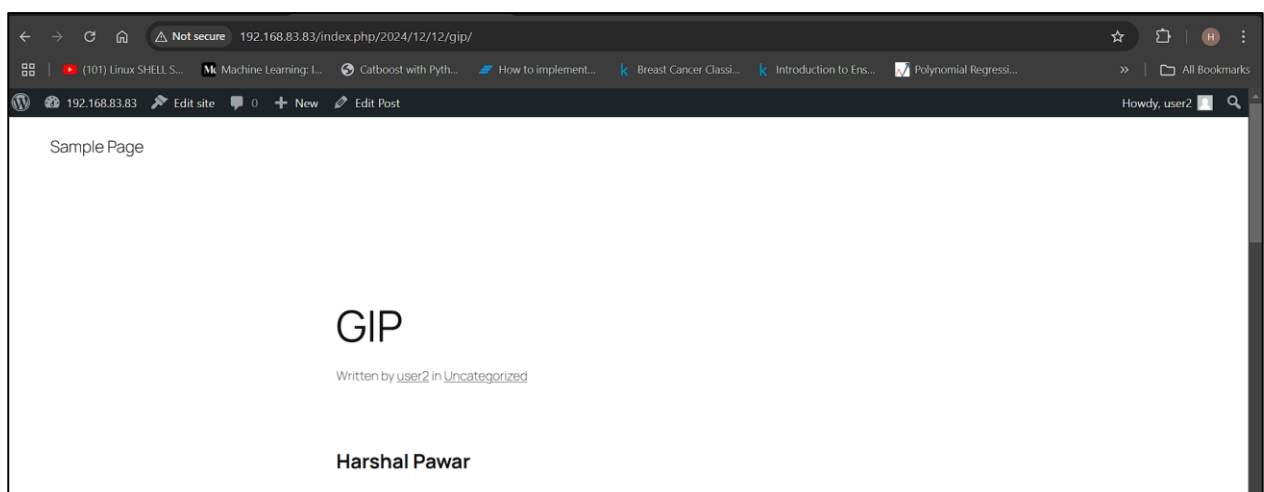
Navigate to Users: Ensure user2 is displayed.

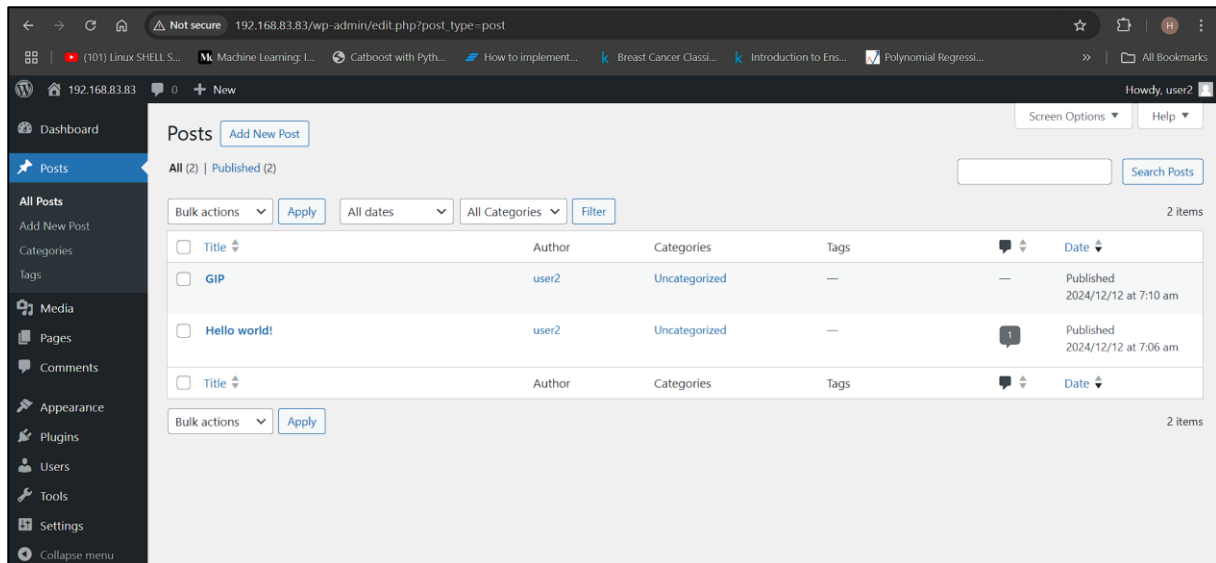


Navigate to Posts: Create a new post and publish it.



Access the post URL to confirm it loads correctly.





From the MariaDB terminal, query the database to confirm posts.

USE wordpress_db;

SELECT ID, post_title, post_date, LEFT(post_content, 100) AS short_content
FROM wp_posts ORDER BY post_date DESC LIMIT 5;

```
MariaDB [wordpress_db]> SELECT ID, post_title, post_date, LEFT(post_content, 100) AS short_content FROM wp_posts ORDER BY post_date DESC LIMIT 5;
+----+-----+-----+-----+
| ID | post_title | post_date | short_content |
+----+-----+-----+-----+
| 6 | GIP | 2024-12-12 07:10:05 | <!-- wp:paragraph {"fontSize":"large"} -->
<p class="has-large-font-size"><strong>Harshal Pawar</str |
| 8 | GIP | 2024-12-12 07:10:05 | <!-- wp:paragraph {"fontSize":"large"} -->
<p class="has-large-font-size"><strong>Harshal Pawar</str |
| 7 | Custom Styles | 2024-12-12 07:08:21 | {"version": 3, "isGlobalStylesUserThemeJSON": true } |
| 5 | Auto Draft | 2024-12-12 07:07:16 | |
| 4 | Navigation | 2024-12-12 07:06:23 | <!-- wp:page-list /--> |
+----+-----+-----+-----+
5 rows in set (0.003 sec)

MariaDB [wordpress_db]> |
```

Troubleshooting

Check if MariaDB is listening on the correct IP and Ensure the firewall allows MySQL traffic.

```
opensuse:~ # ss -tuln | grep 3306
tcp    LISTEN    0      80      0.0.0.0:3306      0.0.0.0:*
opensuse:~ # firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: eth0
  sources:
  services: dhcpv6-client ssh http mysql
  ports: 80/tcp 3260/tcp 3306/tcp
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Verify the Apache configuration file syntax

```
root@ubuntu:~# apache2ctl configtest
Syntax OK
root@ubuntu:~# |
```

Conclusion

The project is successfully integrating MySQL on OpenSUSE with WordPress on Ubuntu, enabling seamless database connectivity for efficient data management and interaction.