

# Deploying Website with Database on AWS Linux server

## Introduction

This document describes the complete process of **hosting and publishing a database-driven website** on **Amazon Web Services (AWS)** using a **Linux (Ubuntu)** server.

The deployed application is a **PHP CRUD (Create, Read, Update, Delete) web application** connected to a **MySQL/MariaDB database**, secured with **HTTPS using Let's Encrypt SSL**, and mapped to a **custom domain (kunjidev.shop)** purchased from **GoDaddy**.

The goal of this task is to demonstrate:

- Cloud server provisioning
- Web server configuration
- Database integration
- Secure deployment using SSL
- Domain and DNS management
- Documentation and testing skills

## Source Code Used

### GitHub Repository:

<https://github.com/Debraj-Dey/CRUD-App-PHP.git>

### Why this source code?

- Open-source and legally usable
- PHP-based web application
- Uses a relational database
- Demonstrates backend functionality
- Ideal for cloud deployment tasks

## Cloud Platform Selection

Platform Used: **Amazon Web Services (AWS)**

**Service:** EC2 (Elastic Compute Cloud)

### Why AWS EC2?

- Industry-standard cloud platform
- Free Tier eligible
- Full control over OS and software
- Supports Linux and Windows
- Scalable and secure

## Server Operating System

**OS: Ubuntu Server 22.04 LTS**

### Why Ubuntu Linux?

- Stable Long Term Support (LTS)
- Widely used in production
- Excellent support for PHP, Nginx, MySQL
- Native Certbot SSL support
- Lightweight and secure

## Architecture Overview

User Browser



Domain (kunjidev.shop)



AWS EC2 (Ubuntu)



Nginx Web Server



PHP-FPM

↓

MySQL / MariaDB Database

## **EC2 Instance Setup (AWS)**

### **Step 1: Launch EC2 Instance**

- AMI: Ubuntu 22.04 LTS
- Instance Type: t2.micro (Free Tier)
- Storage: 8–10 GB
- Key Pair: Created for SSH access
- Security Group:
  - SSH (22) – My IP
  - HTTP (80) – Anywhere
  - HTTPS (443) – Anywhere

### **Step 2: Connect to EC2**

```
ssh -i key.pem ubuntu@<EC2_PUBLIC_IP>
```

**Launch an instance** Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

**Name and tags** Info

**Name**  
My-PHP-Database-Web-Server [Add additional tags](#)

**Application and OS Images (Amazon Machine Image)** Info

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Search our full catalog including 1000s of application and OS images

**Recents** [Quick Start](#)

Amazon Linux macOS Ubuntu Windows Red Hat SUSE Linux Debi

[Browse more AMIs](#)  
Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

[CloudShell](#) [Feedback](#) [Console Mobile App](#)

© 2026, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Account ID: 1295-8554-0940 ▾ MhdAnshad

**Summary**

**Number of instances** [Info](#)  
1

**Software Image (AMI)**  
Canonical, Ubuntu, 24.04, amd64... [read more](#)  
ami-0ecb62995f68bb549

**Virtual server type (instance type)**  
t3.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

**Key pair (login)** Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

**Key pair name - required**  
webserver-KP [Create new key pair](#)

**Network settings** Info

**VPC - required** [Info](#)  
vpc-03f033e27abe565e5 (Default VPC) (default) [Change](#)

**Subnet** [Info](#)  
No preference [Create new subnet](#)

**Availability Zone** [Info](#)  
No preference [Enable additional zones](#)

**Auto-assign public IP** [Info](#)  
Enable

**Firewall (security groups)** [Info](#)  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.  
 Create security group  Select existing security group

[CloudShell](#) [Feedback](#) [Console Mobile App](#)

© 2026, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Account ID: 1295-8554-0940 ▾ MhdAnshad

**Summary**

**Number of instances** [Info](#)  
1

**Software Image (AMI)**  
Canonical, Ubuntu, 24.04, amd64... [read more](#)  
ami-0ecb62995f68bb549

**Virtual server type (instance type)**  
t3.micro

**Firewall (security group)**  
New security group

**Storage (volumes)**  
1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

aws | Search [Alt+S] | United States (N. Virginia) | Account ID: 1295-8554-0940 | MhdAnshad

EC2 > Instances > Launch an instance

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type | Info Protocol | Info Port range | Info  
ssh TCP 22  
Source type | Info Source | Info Description - optional | Info  
Anywhere | Info Add CIDR, prefix list or security group e.g. SSH for admin desktop  
0.0.0.0/0 X Remove

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0)

Type | Info Protocol | Info Port range | Info  
HTTP TCP 80  
Source type | Info Source | Info Description - optional | Info  
Anywhere | Info Add CIDR, prefix list or security group e.g. SSH for admin desktop  
0.0.0.0/0 X Remove

▼ Security group rule 3 (TCP, 443, 0.0.0.0/0)

Type | Info Protocol | Info Port range | Info  
HTTPS TCP 443  
Source type | Info Source | Info Description - optional | Info  
Anywhere | Info Add CIDR, prefix list or security group e.g. SSH for admin desktop  
0.0.0.0/0 X Remove

Summary

Number of instances | Info  
1

Software Image (AMI)  
Canonical, Ubuntu, 24.04, amd64...read more  
ami-0ecb62995f68bb549

Virtual server type (instance type)  
t3.micro

Firewall (security group)  
New security group

Storage (volumes)  
1 volume(s) - 8 GiB

Cancel Launch instance Preview code

CloudShell Feedback Console Mobile App | © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences | Account ID: 1295-8554-0940 | MhdAnshad

EC2 > Instances > Launch an instance

Success  
Successfully initiated launch of instance (i-0f8086a901d2b8b77)

Launch log

Next Steps

Q What would you like to do next with this instance, for example "create alarm" or "create backup"

1 2 3 4 5 6 >

Create billing usage alerts  
To manage costs and avoid surprise bills, set up email notifications for billing usage thresholds.  
Create billing alerts ↗

Connect to your instance  
Once your instance is running, log into it from your local computer.  
Connect to instance ↗  
Learn more ↗

Connect an RDS database  
Configure the connection between an EC2 instance and a database to allow traffic flow between them.  
Connect an RDS database ↗  
Create a new RDS database ↗  
Learn more ↗

Create EBS snapshot policy  
Create a policy that automates the creation, retention, and deletion of EBS snapshots.  
Create EBS snapshot policy ↗

Manage detailed monitoring  
Enable or disable detailed monitoring for

Create Load Balancer  
Create a application, network gateway or

Create AWS budget  
AWS Budgets allows you to create budgets.

Manage CloudWatch alarms  
Create or update Amazon CloudWatch

CloudShell Feedback Console Mobile App | © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```
ubuntu@ip-172-31-75-161: ~ x + v
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\WINDOWS\system32> cd ~/Downloads
PS C:\Users\muham\Downloads> ls

Directory: C:\Users\muham\Downloads

Mode                LastWriteTime         Length Name
----                ——————              ——— ——
d----    05-01-2026 06:25 PM           365 Lounge-1.0.0
d----    05-01-2026 10:39 AM          task
d----    19-12-2025 09:31 PM          Telegram Desktop
d----    09-01-2026 12:14 PM          3132453 Configuring Microsoft 365 Subscription for 10 Users.docx
-a----   05-01-2026 06:25 PM          42128876 godot_spatial_gardener_showcase_windows_x64_v1.4.1-stable.zip
-a----   02-01-2026 12:39 PM          76685794 Godot_v4.5.1-stable_win64.exe.zip
-a----   02-01-2026 11:42 AM          2858111 HOSTING AND PUBLISHING A WEBSITE ON AWS LINUX SERVER.docx
-a----   05-01-2026 04:23 PM          4852112 intern-12.pdf
-a----   03-01-2026 10:28 AM          13356460 INTERNSHIP TASK- 5 CONFIGURING MICROSOFT 365 SUBSCRIPTION FOR USERS.pdf
-a----   05-01-2026 05:04 PM          5327124 Lounge-1.0.0.zip
-a----   05-01-2026 10:39 AM          918 users.csv
-a----   19-12-2025 08:04 PM          1678 webserver-KP.pem
-a----   03-01-2026 10:37 AM          1678 webserver-KP.pem

PS C:\Users\muham\Downloads> ssh -i "webserver-KP.pem" ubuntu@ec2-98-80-140-169.compute-1.amazonaws.com
The authenticity of host 'ec2-98-80-140-169.compute-1.amazonaws.com (98.80.140.169)' can't be established.
ED25519 key fingerprint is SHA256:QZPNbeCGsy5VZkLV5DkZ82s34A1G0wodf3SzYHG4c.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-98-80-140-169.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
```

```
ubuntu@ip-172-31-75-161: ~ x + v
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Wed Jan 14 13:17:53 UTC 2026

System load: 0.22      Temperature:        -273.1 °C
Usage of /: 25.8% of 6.71GB Processes:            116
Memory usage: 25%      Users logged in:     0
Swap usage:  0%          IPv4 address for ens5: 172.31.75.161

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-75-161:~$ |
```

## Web Server Installation

### 7.1 Install Nginx

```
sudo apt update
```

```
sudo apt install nginx -y
```

### Why Nginx?

- High performance
- Low memory usage
- Excellent for PHP websites

- Easy SSL integration

```
ubuntu@ip-172-31-75-161:~$ sudo apt update && sudo apt install nginx -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1692 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [313 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [175 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [15.9 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1510 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [307 kB]
Get:19 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [1404 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [378 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [31.4 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [2426 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [554 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 kB]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [516 kB]
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [30.3 kB]
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [6048 kB]
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 kB]
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [488 kB]
Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Packages [40.4 kB]
Get:31 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main Translation-en [9208 kB]
Get:32 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7308 kB]
Get:33 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [368 kB]
Get:34 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [29.5 kB]
```

Ubuntu Logo

## Apache2 Default Page

**It works!**

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

**Configuration Overview**

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
└── apache2.conf
    ├── ports.conf
    └── mods-enabled
        ├── *.Load
        └── *.conf
    ├── conf-enabled
    └── sites-enabled
        └── *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.

## PHP Installation

```
sudo apt install php php-fpm php-mysql -y
```

### Verify PHP Version

```
php -v
```

### Why PHP-FPM?

- Faster PHP execution
- Secure process isolation
- Recommended for Nginx

```
ubuntu@ip-172-31-75-161:~$ sudo apt install php php-fpm php-mysql -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php8.3 libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64
liblua5.4-0 php-common php8.3 php8.3-cli php8.3-common php8.3-fpm php8.3-mysql php8.3-opcache php8.3-readline ssl-cert
Suggested packages:
apache2-dot apache2-suexec-pristine | apache2-suexec-custom www-browser php-pear
The following NEW packages will be installed:
apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php8.3 libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64
liblua5.4-0 php-common php-fpm php-mysql php8.3 php8.3-cli php8.3-common php8.3-fpm php8.3-mysql php8.3-opcache php8.3-readline ssl-cert
0 upgraded, 22 newly installed, 0 to remove and 82 not upgraded.
Need to get 9062 kB of archives.
After this operation, 36.8 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libapr1t64 amd64 1.7.2-3.1ubuntu0.1 [108 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu8.8 [1331 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 apache2-bin amd64 2.4.58-1ubuntu8.8 [163 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.8 [97.7 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 2.4.58-1ubuntu8.8 [90.2 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 php-common all 2.93ubuntu2 [13.9 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 php8.3-common amd64 8.3.6-0ubuntu0.24.04.6 [748 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 php8.3-opcache amd64 8.3.6-0ubuntu0.24.04.6 [372 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 php8.3-readline amd64 8.3.6-0ubuntu0.24.04.6 [13.4 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 phpc8.3-cgi amd64 8.3.6-0ubuntu0.24.04.6 [1916 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libapache2-mod-php8.3 amd64 8.3.6-0ubuntu0.24.04.6 [1850 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 php8.3-fpm amd64 8.3.6-0ubuntu0.24.04.6 [1925 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 phpc8.3-cgi amd64 8.3.6-0ubuntu0.24.04.6 [9172 B]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 php all 2.8.3+93ubuntu2 [4076 B]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 php-fpm all 2.8.3+93ubuntu2 [4162 B]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 php8.3-mysql amd64 8.3.6-0ubuntu0.24.04.6 [127 kB]
```

```
ubuntu@ip-172-31-75-161:~$ php -v
PHP 8.3.6 (cli) (built: Jan 7 2026 08:40:32) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.6, Copyright (c) Zend Technologies
    with Zend OPcache v8.3.6, Copyright (c), by Zend Technologies
ubuntu@ip-172-31-75-161:~$ |
```

## Database Installation

### Install MariaDB

```
sudo apt install mariadb-server -y
```

### Secure Database

```
sudo mysql_secure_installation
```

## Create Database & User

```
sudo mysql -u root -p
```

```
CREATE DATABASE crud_db;
```

```
CREATE USER 'cruduser'@'localhost' IDENTIFIED BY 'password';
```

```
GRANT ALL PRIVILEGES ON crud_db.* TO 'cruduser'@'localhost';
```

```
FLUSH PRIVILEGES;
```

## Why MariaDB?

- Open-source MySQL compatible
- Faster and secure
- Production ready

```
ubuntu@ip-172-31-75-161:~$ sudo apt install mariadb-server -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
galera-4 libcgifast-perl libcgipm-perl libclone-perl libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libencode-locale-perl libfcgi-bin
libfcgi-perl libfcgi0t64 libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
liblwp-mediatypes-perl libmariadb3 libmysqlclient21 libsnappy1v5 libtimedate-perl liburi-perl liburing2 mariadb-client mariadb-client-core
mariadb-common mariadb-plugin-provider-bzip2 mariadb-plugin-provider-lz4 mariadb-provider-lzma mariadb-plugin-provider-lzo
mariadb-plugin-snappy mariadb-server-core mysql-common socat
Suggested packages:
libltdbm-perl libnet-daemon-perl libsql-statement-perl libdata-dump-perl libipc-sharedcache-perl libio-compress-brotli-perl
libbusiness-isbn-perl libregexp-ipv6-perl libwww-perl mailx mariadb-test doc-base
The following NEW packages will be installed:
galera-4 libcgifast-perl libcgipm-perl libclone-perl libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libencode-locale-perl libfcgi-bin
libfcgi-perl libfcgi0t64 libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl
liblwp-mediatypes-perl libmariadb3 libmysqlclient21 libsnappy1v5 libtimedate-perl liburi-perl liburing2 mariadb-client mariadb-client-core
mariadb-common mariadb-plugin-provider-bzip2 mariadb-plugin-provider-lz4 mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo
mariadb-plugin-snappy mariadb-server mariadb-server-core mysql-common socat
0 upgraded, 37 newly installed, 0 to remove and 82 not upgraded.
Need to get 19.7 MB of archives.
After this operation, 201 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 galera-4 amd64 26.4.16-2build4 [736 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 mysql-common all 5.8+1.1.0build1 [6746 B]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 mariadb-common all 1:10.11.13-0ubuntu0.24.04.1 [28.3 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libdbi-perl amd64 1.643-4build3 [721 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libconfig-inifiles-perl all 3.000003-2 [39.4 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 libmariadb3 amd64 1:10.11.13-0ubuntu0.24.04.1 [196 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 mariadb-client-core amd64 1:10.11.13-0ubuntu0.24.04.1 [1037 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 mariadb-client amd64 1:10.11.13-0ubuntu0.24.04.1 [2500 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liburing2 amd64 2.5-1build1 [21.1 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 mariadb-server-core amd64 1:10.11.13-0ubuntu0.24.04.1 [8392 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 socat amd64 1.8.0.0-4build3 [374 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 mariadb-server amd64 1:10.11.13-0ubuntu0.24.04.1 [3464 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libhtml-tagset-perl all 3.20-6 [11.3 kB]
```

```
ubuntu@ip-172-31-75-161:~$ sudo 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
haven't set the root password yet, you should just press enter here.

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

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

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] n
... skipping.

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] n
... skipping.

Normally, root should only be allowed to connect from 'localhost'. This
```

```
ubuntu@ip-172-31-75-161: ~ + - X
By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] n
... skipping.

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] n
... skipping.

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] n
... skipping.

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] n
... skipping.

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
ubuntu@ip-172-31-75-161:~$ |
```

```
ubuntu@ip-172-31-75-161: ~ + - X
ubuntu@ip-172-31-75-161:~$ sudo mysql
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.11.13-MariaDB-0ubuntu0.24.04.1 Ubuntu 24.04

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)]> CREATE DATABASE crud_db;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> CREATE USER 'cruduser'@'localhost' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.004 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON crud_db.* TO 'cruduser'@'localhost';
Query OK, 0 rows affected (0.002 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> |
```

## Deploy Application Code

### Clone Repository

```
cd /var/www/
```

```
sudo apt install git -y
```

```
sudo git clone https://github.com/Debraj-Dey/CRUD-App-PHP.git crudapp
```

```
sudo chown -R www-data:www-data crudapp
```

### Configure Database in PHP

Update database credentials inside the project configuration file.

```
ubuntu@ip-172-31-75-161:/var/www$ cd /var/www/
ubuntu@ip-172-31-75-161:/var/www$ sudo apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.3).
git set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 82 not upgraded.
ubuntu@ip-172-31-75-161:/var/www$ sudo git clone https://github.com/Debraj-Dey/CRUD-App-PHP.git crudapp
Cloning into 'crudapp'...
remote: Enumerating objects: 32, done.
remote: Counting objects: 100% (32/32), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 32 (delta 10), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (32/32), 13.71 KiB | 4.57 MiB/s, done.
Resolving deltas: 100% (10/10), done.
ubuntu@ip-172-31-75-161:/var/www$ ls
crudapp  html
ubuntu@ip-172-31-75-161:/var/www$ sudo chown -R www-data:www-data crudapp
ubuntu@ip-172-31-75-161:/var/www$ |
```

## Database Schema Setup (MANDATORY)

Check for SQL File

```
ls /var/www/crudapp
```

If .sql file exists:

```
sudo mysql -u root -p crud_db < database.sql
```

If no SQL file exists, manually create table:

```
USE crud_db;
```

```
CREATE TABLE users (
```

```
    id INT AUTO_INCREMENT PRIMARY KEY,
```

```
    name VARCHAR(100),
```

```
    email VARCHAR(100)
```

```
);
```

```

ubuntu@ip-172-31-75-161:~ $ ls /var/www/crudapp
README.md connect.php database.sql delete.php footer.php header.php index.php insert.php script.js style.css update.php
ubuntu@ip-172-31-75-161:~ $ sudo mysql -u root -p crud_db < /var/www/crudapp/database.sql
Enter password:
-----
create database crud
-----

ERROR 1007 (HY000) at line 17: Can't create database 'crud'; database exists
ubuntu@ip-172-31-75-161:~ $ sudo nano /var/www/crudapp/database.sql
ubuntu@ip-172-31-75-161:~ $

```

```

ubuntu@ip-172-31-75-161:~ $ nano 7.2 /var/www/crudapp/database.sql
-- For creating the database
CREATE DATABASE IF NOT EXISTS crud_db;

USE crud_db;

-- For creating the table
CREATE TABLE `users` (
  `id` INT(10) NOT NULL AUTO_INCREMENT,
  `Name` VARCHAR(20) NOT NULL,
  `Gender` ENUM('Male','Female','Other') NOT NULL,
  `Email` VARCHAR(30) NOT NULL,
  `Mobile` VARCHAR(15) NOT NULL,
  `Address` VARCHAR(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE = InnoDB;
-- For creating the database
create database crud;

use crud;

-- For creating the table
CREATE TABLE `users`(`id` INT(10) NOT NULL AUTO_INCREMENT , `Name` VARCHAR(20) NOT NULL , `Gender` ENUM('Male','Female','Other') NOT NULL , `Email` VARCHAR(30) NOT NULL , `Mobile` VARCHAR(15) NOT NULL , `Address` VARCHAR(50) NOT NULL , PRIMARY KEY (`id`)) ENGINE = InnoDB;

```

[ Read 23 Lines (Converted from DOS format) ]

**GNU nano 7.2**

^G Help ^O Write Out ^W Where Is ^T Execute ^C Location M-U Undo M-A Set Mark M-[ To Bracket  
 ^X Exit ^R Read File ^A Replace ^K Cut ^J Justify ^/ Go To Line M-E Redo M-6 Copy M-Q Where Was

## PHP Database Connection Setup (CRITICAL)

### Locate DB Configuration File

```
cd /var/www/crudapp
```

```
grep -R "mysqli" .
```

### Edit Database Config (example: db.php)

```
<?php
```

```
$host = "localhost";
```

```
$user = "cruduser";
```

```

$password = "StrongPassword";

$database = "crud_db";

$conn = mysqli_connect($host, $user, $password, $database);

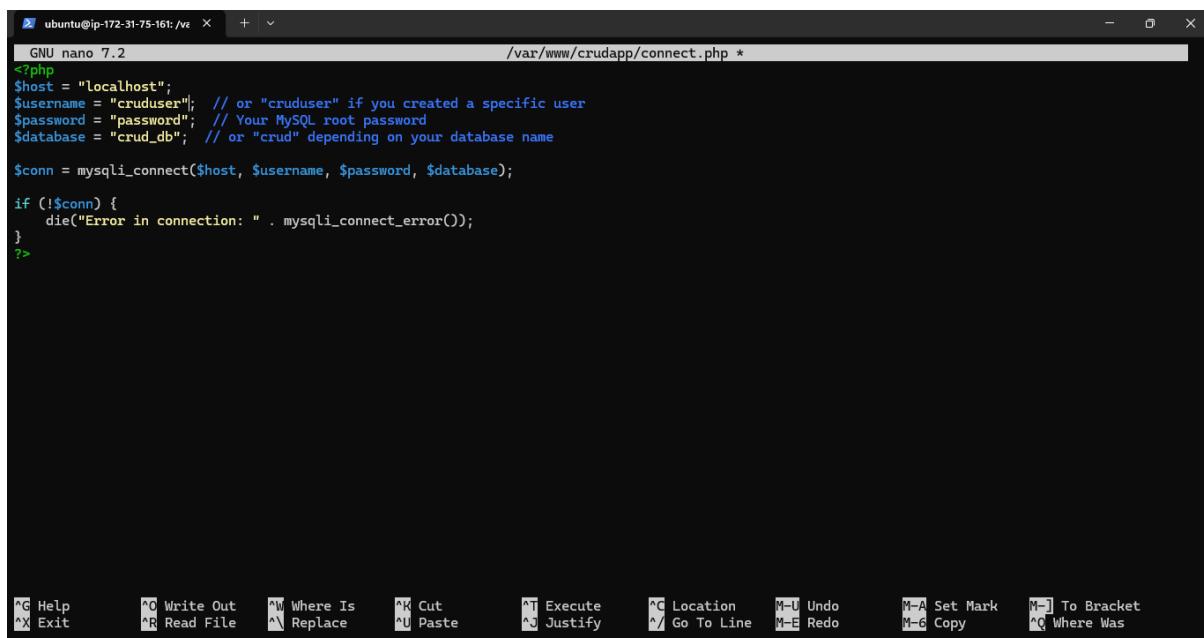
if (!$conn) {
    die("Database connection failed: " . mysqli_connect_error());
}

?>

```

### Why this step is critical?

- PHP does not auto-connect to databases
- Incorrect credentials will break the application



The screenshot shows a terminal window titled 'ubuntu@ip-172-31-75-161: /var/www/crudapp' with the file '/var/www/crudapp/connect.php' open in the nano editor. The code in the editor is:

```

<?php
$host = "localhost";
$username = "cruduser"; // or "cruduser" if you created a specific user
$password = "password"; // Your MySQL root password
$database = "crud_db"; // or "crud" depending on your database name

$conn = mysqli_connect($host, $username, $password, $database);

if (!$conn) {
    die("Error in connection: " . mysqli_connect_error());
}
?>

```

The terminal window includes standard nano key bindings at the bottom.

### Nginx Configuration

```

server {
    listen 80;

    server_name kunjidev.shop www.kunjidev.shop;

```

```
root /var/www/crudapp;  
index index.php index.html;  
  
location / {  
    try_files $uri $uri/ /index.php;  
}  
  
location ~ \.php$ {  
    include snippets/fastcgi-php.conf;  
    fastcgi_pass unix:/var/run/php/php8.1-fpm.sock;  
}  
}
```

Enable site:

```
sudo ln -s /etc/nginx/sites-available/crudapp /etc/nginx/sites-enabled/  
sudo nginx -t  
sudo systemctl reload nginx
```

```
ubuntu@ip-172-31-75-161:~ cd ~
ubuntu@ip-172-31-75-161:~$ sudo nano /etc/nginx/sites-available/crudapp
ubuntu@ip-172-31-75-161:~$ |
```

```
GNU nano 7.2                               /etc/nginx/sites-available/crudapp
server {
    listen 80;
    server_name kunjidev.shop www.kunjidev.shop;

    root /var/www/crudapp;
    index index.php index.html;

    location / {
        try_files $uri $uri/ /index.php;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php8.1-fpm.sock;
    }
}
```

[ Read 16 lines ]  
[ Execute ] [ Location ] [ Undo ] [ Set Mark ] [ To Bracket ]  
[ Justify ] [ Go To Line ] [ Redo ] [ Copy ] [ Where Was ]  
^G Help ^O Write Out ^W Where Is ^K Cut ^U Paste  
^X Exit ^R Read File ^A Replace

```
ubuntu@ip-172-31-75-161:~$ sudo ln -s /etc/nginx/sites-available/crudapp /etc/nginx/sites-enabled/
ubuntu@ip-172-31-75-161:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ip-172-31-75-161:~$ sudo systemctl reload nginx
ubuntu@ip-172-31-75-161:~$ |
```

## Verify PHP Execution

### Create Test File

```
sudo nano /var/www/crudapp/info.php
```

```
<?php phpinfo(); ?>
```

Visit:

[http:// 98.80.140.169/info.php](http://98.80.140.169/info.php)

Confirms PHP is working correctly

The screenshot shows a web browser window displaying the PHP info page. The title bar indicates "PHP 8.3.6 - phpinfo()". The page itself has a blue header with the text "PHP Version 8.3.6" and the PHP logo. Below the header is a table containing detailed PHP configuration information. Key entries include:

System	
Build Date	Jan 7 2026 08:40:32
Build System	Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.3/fpm
Loaded Configuration File	/etc/php/8.3/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.3/fpm/conf.d
Additional .ini files parsed	/etc/php/8.3/fpm/conf.d/10-mysqli.ini, /etc/php/8.3/fpm/conf.d/10-opcache.ini, /etc/php/8.3/fpm/conf.d/10-pdo.ini, /etc/php/8.3/fpm/conf.d/20-calendar.ini, /etc/php/8.3/fpm/conf.d/20-clype.ini, /etc/php/8.3/fpm/conf.d/20-ext.ini, /etc/php/8.3/fpm/conf.d/20-fl_ini.ini, /etc/php/8.3/fpm/conf.d/20-fleinfo.ini, /etc/php/8.3/fpm/conf.d/20-ftp.ini, /etc/php/8.3/fpm/conf.d/20-gettext.ini, /etc/php/8.3/fpm/conf.d/20-iconv.ini, /etc/php/8.3/fpm/conf.d/20-mysqli.ini, /etc/php/8.3/fpm/conf.d/20-pdo_mysql.ini, /etc/php/8.3/fpm/conf.d/20-phar.ini, /etc/php/8.3/fpm/conf.d/20-posix.ini, /etc/php/8.3/fpm/conf.d/20-readline.ini, /etc/php/8.3/fpm/conf.d/20-shmop.ini, /etc/php/8.3/fpm/conf.d/20-sockets.ini, /etc/php/8.3/fpm/conf.d/20-sysvmsg.ini, /etc/php/8.3/fpm/conf.d/20-sysvsem.ini, /etc/php/8.3/fpm/conf.d/20-sysvshm.ini, /etc/php/8.3/fpm/conf.d/20-tokenizer.ini
PHP API	
PHP Extension	20230831
Zend Extension	420230831
Zend Extension Build	API420230831.NTS
PHP Extension Build	API20230831.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled

## **Domain Configuration (GoDaddy)**

**Domain: kunjidev.shop**

**DNS Settings:**

**Type Name Value**

A @ EC2 Public IP

A www EC2 Public IP

**Why DNS mapping?**

- Connects domain name to server IP
- Makes website publicly accessible

[Dashboard](#)

[Domain](#)

[Website](#)

[Email](#)

[Store](#)

[Appointments](#)

[Marketing](#)

[Conversations](#)

[Customers](#)

[Deals](#)

## Kunjidev

kunjidev.shop

[Add Logo](#)

**Everything generated for you by GoDaddy Airo® \***

**Activate your professional email address**

83% of small businesses say a domain-based email helped them look more professional.\*\*

[Activate Email](#)

[Muha...@kunjidevs...](#)

**Customize a website for Kunjidev**

[View Site](#)

\*If you publish a website or generate content using any of the AI enabled tools available here you agree to the [AI Terms of Use](#), [Universal Terms of Service](#) and [Website Services Agreement](#).

Transferring data from dashboard.godaddy.com...

**DNS Records**

**Add a new record**

DNS records define how your domain behaves, like showing your website content and delivering your email.

[Add New Record](#)

**Easily verify domain ownership**

Need to verify ownership of your domain to connect to an external service? We've made it easier than ever.

[Verify Domain Ownership](#)

**Create MX records**

Quickly create MX records to connect your domain with email services.

[Create Now](#)

**New Records**

A records use an IP address to connect your domain to a website. They're also used to [create subdomains](#) such as www or store, that point to an IP address.

Type *	Name *	Value *	TTL
A	@	3.95.201.52	Custom Seconds 600
<a href="#">+ Add another value</a>			

[Save](#) [Cancel](#)

Type <small>(?)</small>	Name <small>(?)</small>	Data <small>(?)</small>	TTL <small>(?)</small>	Delete	Edit

	type	name	data	TTL	Delete	Edit
	A	@	98.80.140.169	600 seconds		
	A	www	98.80.140.169	600 seconds		
	NS	@	ns59.domaincontrol.com.	1 hour	Can't remove	Can't edit
	NS	@	ns60.domaincontrol.com.	1 hour	Can't remove	Can't edit
	CNAME	autodiscover	autodiscover.outlook.com.	1 hour		

## SSL Configuration (HTTPS)

### Install Certbot

```
sudo apt install certbot python3-certbot-nginx -y
```

### Generate SSL Certificate

```
sudo certbot --nginx -d kunjidev.shop -d www.kunjidev.shop
```

### Why SSL?

- Encrypts data
- Improves security
- Required for modern browsers
- SEO ranking benefit

```

ubuntu@ip-172-31-75-161:~ % sudo apt install certbot python3-certbot-nginx -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-acme python3-certbot python3-configargparse python3-icu python3-josepy python3-parsedatetime python3-rfc3339
Suggested packages:
  python-certbot-doc python3-certbot-apache python-acme-doc python-certbot-nginx-doc
The following NEW packages will be installed:
  certbot python3-acme python3-certbot python3-certbot-nginx python3-configargparse python3-icu python3-josepy python3-parsedatetime
  python3-rfc3339
0 upgraded, 9 newly installed, 0 to remove and 84 not upgraded.
Need to get 1097 kB of archives.
After this operation, 5699 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-josepy all 1.14.0-1 [22.1 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-rfc3339 all 1.1-4 [6744 B]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-acme all 2.9.0-1 [48.5 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-configargparse all 1.7-1 [31.7 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-parsedatetime all 2.6-3 [32.8 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-certbot all 2.9.0-1 [267 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 certbot all 2.9.0-1 [89.2 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-certbot-nginx all 2.9.0-1 [66.0 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 python3-icu amd64 2.12-1build2 [534 kB]
Fetched 1097 kB in 0s (25.6 MB/s)
Preconfiguring packages ...
Selecting previously unselected package python3-josepy.
(Reading database ... 73631 files and directories currently installed.)
Preparing to unpack .../0-python3-josepy_1.14.0-1_all.deb ...
Unpacking python3-josepy (1.14.0-1) ...
Selecting previously unselected package python3-rfc3339.
Preparing to unpack .../1-python3-rfc3339_1.1-4_all.deb ...
Unpacking python3-rfc3339 (1.1-4) ...
Selecting previously unselected package python3-acme.
Preparing to unpack .../2-python3-acme_2.9.0-1_all.deb ...
Unpacking python3-acme (2.9.0-1) ...

ubuntu@ip-172-31-75-161:~ % sudo certbot --nginx -d kunjidev.shop -d www.kunjidev.shop
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
(Enter 'c' to cancel): mhdanu247@gmail.com

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.6-August-18-2025.pdf. You must agree
in order to register with the ACME server. Do you agree?
-----
(Y)es/(N)o: y

-----
Would you be willing, once your first certificate is successfully issued, to
share your email address with the Electronic Frontier Foundation, a founding
partner of the Let's Encrypt project and the non-profit organization that
develops Certbot? We'd like to send you email about our work encrypting the web,
EFF news, campaigns, and ways to support digital freedom.
-----
(Y)es/(N)o: y
Account registered.
Requesting a certificate for kunjidev.shop and www.kunjidev.shop

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/kunjidev.shop/fullchain.pem
Key is saved at: /etc/letsencrypt/live/kunjidev.shop/privkey.pem
This certificate expires on 2026-04-14.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for kunjidev.shop to /etc/nginx/sites-enabled/crudapp
Successfully deployed certificate for www.kunjidev.shop to /etc/nginx/sites-enabled/crudapp
Congratulations! You have successfully enabled HTTPS on https://kunjidev.shop and https://www.kunjidev.shop

```

## Testing

### Tests Performed:

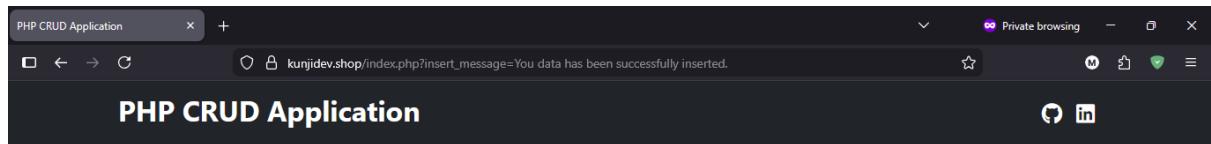
- Website loads via HTTP & HTTPS
- Database CRUD operations verified
- Mobile and desktop browser tested
- SSL certificate validated
- Domain resolution verified

## Live URL

🔗 <https://kunjidev.shop>

The screenshot shows a browser window titled "PHP CRUD Application". The address bar displays "https://kunjidev.shop/". The main content area has a dark header with the title "PHP CRUD Application". Below the header is a table with columns: ID, Name, Gender, Email, Mobile, Address, and Operations. A button labeled "New users" with a plus sign and a user icon is visible. The table currently contains no data.

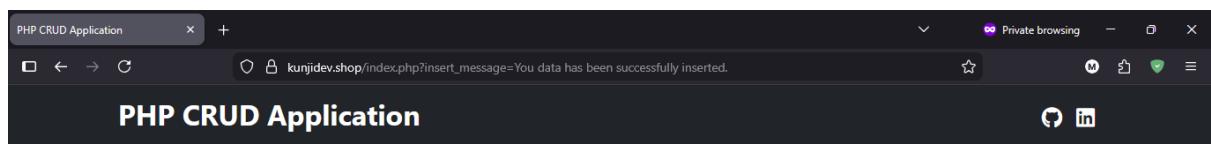
The screenshot shows a modal dialog titled "Add new users" overlaid on the "PHP CRUD Application" page. The dialog contains fields for Name (Anshad), Gender (Male), Email (mhdanu247@gmail.com), Mobile (1234324536), and Address (Thayill House). At the bottom of the dialog are two buttons: "Close" and "Add changes". The background of the application is dimmed.



**New users**

ID	Name	Gender	Email	Mobile	Address	Operations
1	Anshad	Male	mhdanu247@gmail.com	1234324536	Thayyil House	

You data has been successfully inserted.



**New users**

ID	Name	Gender	Email	Mobile	Address	Operations
1	Anshad	Male	mhdanu247@gmail.com	1234324536	Thayyil House	
2	Mazin	Male	mmani2323@gmail.com	4563723645	Thayyil House	

You data has been successfully inserted.

The screenshot shows a browser window titled "PHP CRUD Application". The URL is "kunjidev.shop/index.php?deletion\_message=Data is successfully deleted !!". The page header has "PHP CRUD Application" and social media links for GitHub and LinkedIn. Below the header is a button labeled "New users". A table displays user data with columns: ID, Name, Gender, Email, Mobile, Address, and Operations. One row is shown: ID 1, Name Anshad, Gender Male, Email mhdanu247@gmail.com, Mobile 1234324536, Address Thayyil House, and Operations with edit and delete icons. A red success message "Data is successfully deleted !!" is displayed below the table.

ID	Name	Gender	Email	Mobile	Address	Operations
1	Anshad	Male	mhdanu247@gmail.com	1234324536	Thayyil House	

Data is successfully deleted !!

## Challenges & Solutions

### Challenge      Solution

PHP not loading Configured PHP-FPM

Database error    Corrected credentials

SSL failure        Opened port 443

DNS delay         Waited for propagation

---

### Additional Insights

- Linux servers are ideal for PHP workloads
- Nginx + PHP-FPM provides high performance
- Let's Encrypt SSL simplifies HTTPS
- AWS Free Tier is sufficient for learning

---

### Conclusion

This project successfully demonstrates **end-to-end deployment of a database-driven website** on a **cloud infrastructure** using **best industry practices**.

The application is **secure, scalable, and publicly accessible**, fulfilling all requirements of the task.

The deployment reflects real-world DevOps and cloud hosting workflows and provides a strong foundation for production-ready applications.