

PANDUAN DEPLOY KE VPS UBUNTU

Production Deployment Guide

Ubuntu 22.04/24.04 LTS

JagoFlutter Academy - AFC6

Apotek POS System Fullstack

Flutter 3.38 • Laravel 12 • Filament 4

DAFTAR ISI

- 1. Persyaratan Server
- 2. Persiapan VPS
- 3. Instalasi Software
- 4. Setup Database
- 5. Deploy Aplikasi Laravel
- 6. Konfigurasi Nginx
- 7. Setup SSL/HTTPS
- 8. Optimasi Production
- 9. Maintenance & Backup
- 10. Troubleshooting

1. PERSYARATAN SERVER

Minimum Specifications:

Komponen	Spesifikasi
OS	Ubuntu 22.04 LTS atau 24.04 LTS
RAM	2 GB (4 GB recommended)
CPU	2 vCPU
Storage	20 GB SSD
Bandwidth	Unlimited atau minimal 1 TB/bulan

Provider VPS Recommended:

- bullet Biznet GioCloud
- bullet DigitalOcean
- bullet Vultr
- bullet Linode
- bullet AWS Lightsail
- bullet Google Cloud Platform

2. PERSIAPAN VPS

Langkah 1: Login ke VPS

```
ssh username@ip_public_vps

# Contoh:
ssh root@103.123.456.789
```

Langkah 2: Update System

```
sudo apt update && sudo apt upgrade -y
```

Langkah 3: Setup Timezone

```
sudo timedatectl set-timezone Asia/Jakarta

# Verifikasi
date
```

Langkah 4: Buat User Non-Root (Recommended)

```
# Buat user baru
sudo adduser deployer

# Tambahkan ke sudo group
sudo usermod -aG sudo deployer

# Switch ke user baru
su - deployer
```

3. INSTALASI SOFTWARE

1. Install Nginx

```
sudo apt install nginx -y

# Start dan enable Nginx
sudo systemctl start nginx
sudo systemctl enable nginx

# Cek status
sudo systemctl status nginx
```

TIP: Verifikasi: Buka http://ip_vps_anda di browser, akan muncul halaman default Nginx.

2. Install MariaDB

```
# Install MariaDB Server dan Client
sudo apt install mariadb-server mariadb-client -y

# Start dan enable MariaDB
sudo systemctl start mariadb
sudo systemctl enable mariadb

# Secure installation
sudo mysql_secure_installation
```

Saat mysql_secure_installation:

- bullet Enter current password for root: (tekan Enter jika kosong)
- bullet Switch to unix_socket authentication: N
- bullet Change the root password: Y (masukkan password baru)
- bullet Remove anonymous users: Y
- bullet Disallow root login remotely: Y
- bullet Remove test database: Y
- bullet Reload privilege tables: Y

3. Install PHP 8.3

```
# Install dependencies
sudo apt install software-properties-common -y

# Tambah repository PHP
sudo add-apt-repository ppa:ondrej/php -y
sudo apt update

# Install PHP 8.3 dengan extensions
sudo apt install php8.3-fpm php8.3-cli php8.3-common php8.3-mysql \
    php8.3-zip php8.3-gd php8.3-mbstring php8.3-curl php8.3-xml \
    php8.3-bcmath php8.3-intl php8.3-readline php8.3-redis -y

# Verifikasi
php -v
```

4. Install Composer

```
cd ~
curl -sS https://getcomposer.org/installer -o composer-setup.php
sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer
rm composer-setup.php

# Verifikasi
composer -V
```

5. Install Node.js dan NPM

```
curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
sudo apt install nodejs -y

# Verifikasi
node -v
npm -v
```

6. Install Git

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

4. SETUP DATABASE

Buat Database dan User:

```
# Login ke MariaDB
sudo mysql -u root -p

-- Buat database
CREATE DATABASE apotek_pos CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;

-- Buat user database
CREATE USER 'apotek_user'@'localhost' IDENTIFIED BY 'password_kuat_anda';

-- Berikan privileges
GRANT ALL PRIVILEGES ON apotek_pos.* TO 'apotek_user'@'localhost';

-- Apply privileges
FLUSH PRIVILEGES;

-- Keluar
EXIT;
```

Test Koneksi:

```
mysql -u apotek_user -p apotek_pos
```

5. DEPLOY APLIKASI LARAVEL

Langkah 1: Clone Repository

```
cd /var/www
sudo git clone https://github.com/[username]/laravel-apotik-be.git apotek
cd apotek
sudo chown -R $USER:www-data .
```

Langkah 2: Install Dependencies

```
composer install --optimize-autoloader --no-dev
npm install
npm run build
```

Langkah 3: Setup Environment

```
cp .env.example .env
nano .env
```

Konfigurasi .env untuk Production:

```
APP_NAME="Apotek POS"
APP_ENV=production
APP_KEY=
APP_DEBUG=false
APP_URL=https://apotek.domain-anda.com

LOG_CHANNEL=stack
LOG_LEVEL=error

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=apotek_pos
DB_USERNAME=apotek_user
DB_PASSWORD=password_kuat_anda

CACHE_DRIVER=file
SESSION_DRIVER=file
QUEUE_CONNECTION=database
```

Langkah 4: Generate Key dan Migrasi

```
php artisan key:generate
php artisan migrate --force
php artisan db:seed --force
php artisan storage:link
```

Langkah 5: Set Permissions

```
sudo chmod -R 775 storage bootstrap/cache
sudo chown -R $USER:www-data storage bootstrap/cache
```

Langkah 6: Optimasi Laravel

```
php artisan config:cache
php artisan route:cache
php artisan view:cache
composer dump-autoload --optimize
```

6. KONFIGURASI NGINX

Buat Virtual Host:

```
sudo nano /etc/nginx/sites-available/apotek
```

Isi Konfigurasi:

```
server {
    listen 80;
    listen [::]:80;
    server_name apotek.domain-anda.com;
    root /var/www/apotek/public;

    add_header X-Frame-Options "SAMEORIGIN";
    add_header X-Content-Type-Options "nosniff";

    index index.php;
    charset utf-8;

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

    location = /favicon.ico { access_log off; log_not_found off; }
    location = /robots.txt  { access_log off; log_not_found off; }

    error_page 404 /index.php;

    location ~ \.php$ {
        fastcgi_pass unix:/var/run/php/php8.3-fpm.sock;
        fastcgi_param SCRIPT_FILENAME $realpath_root$fastcgi_script_name;
        include fastcgi_params;
    }

    location ~ /\.(!well-known).* {
        deny all;
    }

    client_max_body_size 50M;
}
```

Enable Site:

```
# Buat symbolic link
sudo ln -s /etc/nginx/sites-available/apotek /etc/nginx/sites-enabled/

# Hapus default site (opsional)
sudo rm /etc/nginx/sites-enabled/default

# Test konfigurasi
sudo nginx -t

# Restart Nginx
sudo systemctl restart nginx
```

7. SETUP SSL/HTTPS

Install Certbot (Let's Encrypt):

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

# Dapatkan SSL Certificate
sudo certbot --nginx -d apotek.domain-anda.com
```

Ikuti instruksi:

- bullet Masukkan email address
- bullet Agree to terms of service: Y
- bullet Share email with EFF: N (opsional)
- bullet Redirect HTTP to HTTPS: 2 (recommended)

Verifikasi Auto-Renewal:

```
sudo certbot renew --dry-run
sudo systemctl list-timers | grep certbot
```

Update .env:

```
nano /var/www/apotek/.env
# Ubah: APP_URL=https://apotek.domain-anda.com

php artisan config:cache
```

8. OPTIMASI PRODUCTION

1. Setup Queue Worker dengan Supervisor

```
sudo apt install supervisor -y
sudo nano /etc/supervisor/conf.d/apotek-worker.conf
```

Isi Konfigurasi:

```
[program:apotek-worker]
process_name=%(program_name)s_%(process_num)02d
command=php /var/www/apotek/artisan queue:work --sleep=3 --tries=3 --max-time=3600
autostart=true
autorestart=true
stopasgroup=true
killasgroup=true
user=www-data
numprocs=2
redirect_stderr=true
stdout_logfile=/var/www/apotek/storage/logs/worker.log
stopwaitsecs=3600
```

```
sudo supervisorctl reread
sudo supervisorctl update
sudo supervisorctl start apotek-worker:*
```

2. Setup Cron untuk Scheduler

```
crontab -e

# Tambahkan:
* * * * * cd /var/www/apotek && php artisan schedule:run >> /dev/null 2>&1
```

3. Konfigurasi Firewall (UFW)

```
sudo ufw enable
sudo ufw allow OpenSSH
sudo ufw allow 'Nginx Full'
sudo ufw status
```

4. Optimasi PHP-FPM

```
sudo nano /etc/php/8.3/fpm/php.ini

# Sesuaikan:
memory_limit = 256M
upload_max_filesize = 50M
post_max_size = 50M
max_execution_time = 60

# Restart
sudo systemctl restart php8.3-fpm
```

9. MAINTENANCE & BACKUP

Script Backup Otomatis:

```
#!/bin/bash
BACKUP_DIR="/var/www/backups"
APP_DIR="/var/www/apotek"
DB_NAME="apotek_pos"
DB_USER="apotek_user"
DB_PASS="password_anda"
DATE=$(date +%Y%m%d_%H%M%S)

mkdir -p $BACKUP_DIR

# Backup database
mysqldump -u $DB_USER -p$DB_PASS $DB_NAME > $BACKUP_DIR/db_$DATE.sql
gzip $BACKUP_DIR/db_$DATE.sql

# Backup storage files
tar -czf $BACKUP_DIR/storage_$DATE.tar.gz -C $APP_DIR storage

# Hapus backup lebih dari 7 hari
find $BACKUP_DIR -type f -mtime +7 -delete

echo "Backup completed: $DATE"
```

Set Permissions dan Jadwalkan:

```
sudo chmod +x /var/www/backup-apotek.sh

# Tambah ke crontab (backup setiap hari jam 2 pagi)
crontab -e
0 2 * * * /var/www/backup-apotek.sh >> /var/log/backup-apotek.log 2>&1
```

10. TROUBLESHOOTING

Error 502 Bad Gateway

```
# Penyebab: PHP-FPM tidak berjalan
sudo systemctl status php8.3-fpm
sudo systemctl restart php8.3-fpm
sudo systemctl restart nginx
```

Error 500 Internal Server Error

```
# Cek log
tail -f /var/www/apotek/storage/logs/laravel.log
tail -f /var/log/nginx/error.log

# Fix permissions
sudo chmod -R 775 /var/www/apotek/storage
sudo chown -R www-data:www-data /var/www/apotek/storage
```

SSL Certificate Tidak Renew

```
sudo certbot renew --dry-run
sudo certbot renew --force-renewal
sudo cat /var/log/letsencrypt/letsencrypt.log
```

Queue Worker Tidak Berjalan

```
sudo supervisorctl status
sudo supervisorctl restart apotek-worker:*
tail -f /var/www/apotek/storage/logs/worker.log
```

QUICK COMMANDS REFERENCE

```
# Start/Stop/Restart Services
sudo systemctl start nginx
sudo systemctl restart php8.3-fpm
sudo systemctl restart mariadb

# Laravel Artisan
php artisan down                # Maintenance mode ON
php artisan up                  # Maintenance mode OFF
php artisan migrate --force      # Run migrations
php artisan config:cache         # Cache config
php artisan route:cache          # Cache routes
php artisan view:cache           # Cache views
php artisan optimize:clear       # Clear all cache

# Supervisor
sudo supervisorctl status
sudo supervisorctl restart all

# Logs
tail -f /var/www/apotek/storage/logs/laravel.log
```

DEPLOYMENT CHECKLIST

- VPS sudah disiapkan dengan Ubuntu 22.04/24.04
- Nginx terinstall dan berjalan
- MariaDB terinstall dan dikonfigurasi
- PHP 8.3 dengan semua extensions terinstall
- Composer terinstall
- Node.js dan NPM terinstall
- Git terinstall
- Database sudah dibuat
- Repository sudah di-clone
- Dependencies terinstall (composer install, npm install)
- File .env sudah dikonfigurasi
- Application key sudah di-generate
- Migrasi database sudah dijalankan
- Assets sudah di-build
- Storage link sudah dibuat
- Permissions sudah diset dengan benar
- Virtual host Nginx sudah dikonfigurasi
- SSL certificate sudah terpasang
- Firewall sudah dikonfigurasi
- Queue worker sudah disetup dengan Supervisor
- Cron job untuk scheduler sudah disetup
- Script backup sudah disiapkan