

# Contents

<b>INSTALLATION GUIDE</b>	<b>1</b>
Prerequisites . . . . .	1
Software Requirements . . . . .	1
PHP Extensions Required . . . . .	1
Server Requirements . . . . .	2
Installation Steps . . . . .	2
1. Clone Repository . . . . .	2
2. Install PHP Dependencies . . . . .	2
3. Environment Configuration . . . . .	2
4. Create Database . . . . .	3
5. Run Migrations & Seeders . . . . .	3
6. Storage Symlink . . . . .	4
7. Install Frontend Dependencies . . . . .	4
8. Compile Assets . . . . .	4
9. Cache Configuration (Production Only) . . . . .	4
10. Set Permissions (Linux/Mac) . . . . .	4
11. Start Development Server . . . . .	4
Default Login Credentials . . . . .	5
Additional Setup . . . . .	5
Queue Worker (Background Jobs) . . . . .	5
Task Scheduler (Cron Jobs) . . . . .	5
Redis Cache (Optional) . . . . .	5
Verification Checklist . . . . .	6
Troubleshooting . . . . .	6
Next Steps . . . . .	6

## INSTALLATION GUIDE

### Prerequisites

Sebelum instalasi, pastikan sistem Anda memiliki:

#### Software Requirements

Software	Minimum Version	Recommended
<b>PHP</b>	8.2	8.3
<b>Composer</b>	2.5	Latest
<b>MySQL / MariaDB</b>	8.0 / 10.6	Latest
<b>Node.js</b>	18.x	20.x LTS
<b>NPM</b>	9.x	Latest
<b>Git</b>	2.x	Latest

#### PHP Extensions Required

```
# Check PHP extensions
php -m
```

```
# Required extensions:
- OpenSSL
- PDO
- PDO_MySQL
```

- Mbstring
- Tokenizer
- XML
- Ctype
- JSON
- BCMath
- GD atau Imagick
- Zip
- Fileinfo

## Server Requirements

**Development:** - RAM: 2GB minimum - Storage: 5GB free space - OS: Windows/Mac/Linux

**Production:** - RAM: 4GB recommended - Storage: 50GB SSD - OS: Ubuntu 22.04 LTS / Debian 11+

---

## Installation Steps

### 1. Clone Repository

```
# Via HTTPS
git clone https://github.com/your-org/koperasi.git

# Via SSH (jika sudah setup)
git clone git@github.com:your-org/koperasi.git

# Navigate to project directory
cd koperasi
```

### 2. Install PHP Dependencies

```
# Install via Composer
composer install

# Untuk production (tanpa dev dependencies)
composer install --no-dev --optimize-autoloader
```

**Common Issues:** - Jika error composer command not found, install Composer terlebih dahulu - Jika error memory limit, tambahkan COMPOSER\_MEMORY\_LIMIT=-1 sebelum command

### 3. Environment Configuration

```
# Copy .env example
cp .env.example .env

# Generate application key
php artisan key:generate
```

Edit .env file:

```
# Application
APP_NAME="Koperasi Karyawan"
APP_ENV=local # production untuk server live
APP_DEBUG=true # false untuk production
APP_URL=http://localhost:8000
APP_TIMEZONE=Asia/Jakarta
```

```

# Database
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=koperasi
DB_USERNAME=root
DB_PASSWORD=your_password_here

# Mail (SMTP)
MAIL_MAILER=smtp
MAIL_HOST=smtp.gmail.com
MAIL_PORT=587
MAIL_USERNAME=your_email@gmail.com
MAIL_PASSWORD=your_app_password
MAIL_ENCRYPTION=tls
MAIL_FROM_ADDRESS=noreply@koperasi.com
MAIL_FROM_NAME="${APP_NAME}"

# Midtrans (Payment Gateway)
MIDTRANS_SERVER_KEY=your_server_key
MIDTRANS_CLIENT_KEY=your_client_key
MIDTRANS_IS_PRODUCTION=false # true untuk production

# WhatsApp API (Fonnte)
WHATSAPP_API_TOKEN=your_fonnte_token
WHATSAPP_SENDER=6281234567890

```

#### 4. Create Database

Via MySQL Command:

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

```
CREATE DATABASE koperasi CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
exit;
```

Via phpMyAdmin: 1. Buka phpMyAdmin (<http://localhost/phpmyadmin>) 2. Klik “New” di sidebar 3. Nama database: koperasi 4. Collation: utf8mb4\_unicode\_ci 5. Klik “Create”

#### 5. Run Migrations & Seeders

```
# Fresh install (reset semua tabel + seed data)
php artisan migrate:fresh --seed
```

```
# Atau migrate saja (tanpa delete data)
php artisan migrate
```

```
# Seed data demo (opsional)
php artisan db:seed
```

**Seeder akan membuat:** - User admin default - Sample members - Product categories - Sample products - Chart of accounts

## 6. Storage Symlink

```
# Create symbolic link  
php artisan storage:link
```

**Windows Users:** Jika error “symlink(): Protocol error”, jalankan CMD/PowerShell as **Administrator**

## 7. Install Frontend Dependencies

```
# Install NPM packages  
npm install
```

```
# Atau dengan Yarn  
yarn install
```

## 8. Compile Assets

```
# Development (dengan watch untuk auto-compile)  
npm run dev
```

```
# Production (minified & optimized)  
npm run build
```

## 9. Cache Configuration (Production Only)

```
# Cache config  
php artisan config:cache
```

```
# Cache routes  
php artisan route:cache
```

```
# Cache views  
php artisan view:cache
```

```
# Optimize autoloader  
composer install --optimize-autoloader --no-dev
```

## 10. Set Permissions (Linux/Mac)

```
# Laravel storage & cache writable  
sudo chmod -R 775 storage bootstrap/cache
```

```
# Set ownership (ganti 'www-data' sesuai web server user)  
sudo chown -R www-data:www-data storage bootstrap/cache
```

## 11. Start Development Server

```
# Laravel built-in server  
php artisan serve
```

```
# Access at: http://localhost:8000
```

```
# Custom host & port  
php artisan serve --host=192.168.1.100 --port=8080
```

---

## Default Login Credentials

**Admin Account:** - Email: admin@koperasi.local - Password: password

**Pengurus Account:** - Email: pengurus@koperasi.local - Password: password

**Member Account:** - Email: member@koperasi.local - Password: password

**PENTING:** Ganti password default setelah login!

---

## Additional Setup

### Queue Worker (Background Jobs)

```
# Start queue worker
php artisan queue:work

# Dengan supervisor (production)
# Edit /etc/supervisor/conf.d/laravel-worker.conf
[program:laravel-worker]
process_name=%(program_name)s_%(process_num)02d
command=php /var/www/koperasi/artisan queue:work --sleep=3 --tries=3
autostart=true
autorestart=true
user=www-data
numprocs=2
redirect_stderr=true
stdout_logfile=/var/www/koperasi/storage/logs/worker.log
```

### Task Scheduler (Cron Jobs)

Linux/Mac:

```
# Edit crontab
crontab -e

# Add this line:
* * * * * cd /path-to-your-project && php artisan schedule:run >> /dev/null 2>&1
```

Windows: Gunakan Task Scheduler untuk run command setiap menit.

### Redis Cache (Optional)

```
# Install Redis server

# Install PHP Redis extension
pecl install redis

# Update .env
CACHE_DRIVER=redis
SESSION_DRIVER=redis
QUEUE_CONNECTION=redis

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379
```

---

## Verification Checklist

Setelah instalasi, pastikan:

- Aplikasi bisa diakses via browser
- Login dengan akun default berhasil
- Dashboard tampil dengan benar
- Upload foto profil berfungsi
- Email test terkirim (cek SMTP)
- Queue worker berjalan
- Cron job aktif (cek log)

### Test Command:

```
# Send test email
php artisan tinker
Mail::raw('Test email from Koperasi App', function($message) {
    $message->to('your-email@example.com')->subject('Test Email');
});
exit;
```

---

## Troubleshooting

Lihat TROUBLESHOOTING.md untuk solusi common issues.

### Quick Fixes:

```
# Clear all cache
php artisan cache:clear
php artisan config:clear
php artisan route:clear
php artisan view:clear

# Recreate autoload files
composer dump-autoload

# Fix storage permissions
chmod -R 775 storage bootstrap/cache
```

---

## Next Steps

1. Ganti password default
2. Setup SMTP email
3. Konfigurasi payment gateway
4. Setup backup schedule
5. Customize branding (logo, nama koperasi)
6. Import data anggota
7. Test semua fitur
8. Deploy ke production server

---

**Need Help?** - Email: support@kopkarskf.com - Full Docs: FEATURES.md