Contents

1   INSTALLATION GUIDE

1.1   Prerequisites

Sebelum instalasi, pastikan sistem Anda memiliki:

1.1.1   Software Requirements

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

1.1.2   PHP Extensions Required

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

Daftar Isi

1 fi PETUNJUK INSTALASI

1.1 Prasyarat

Sebelum instalasi, pastikan sistem Anda memiliki:

1.1.1 Persyaratan Perangkat Lunak

Perangkat Lunak	Versi Minimum	yang Direkomendasikan
PHP	8.2	8.3
ComposerMySQL / MariaDBNode.js	2.5	Terbaru
	8.0 / 10.6	Terbaru
	18.	20.x LTS
NPM	x9.x	Terbaru
Git	2.x	Terbaru

1.1.2 Ekstensi PHP yang Diperlukan

```
# Periksa ekstensi PHP
php -m
```

```
# Required extensions:
- OpenSSL
- PDO
- PDO_MySQL
- Mbstring
- Tokenizer
- XML
- Ctype
- JSON
- BCMath
- GD atau Imagick
- Zip
- Fileinfo
```

1.1.3 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+

1.2 Installation Steps

1.2.1 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
```

1.2.2 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

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

1.1.3 Persyaratan Server

Pengembangan: - RAM: 2GB minimum - Penyimpanan: 5GB ruang kosong - OS: Windows/Mac/LinuxProduksi: - RAM: 4GB disarankan - Penyimpanan: 50GB SSD - OS: Ubuntu 22.04 LTS / Debian11+

1.2 Langkah Instalasi

1.2.1 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

# Navigasi ke direktori proyek
cd koperasi
```

1.2.2 2. Instal Ketergantungan PHP

```
# Instal melalui Composer
composer install

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

Masalah Umum: - Jika error composer command not found, instal Composer terlebih dahulu - Jika error memory limit, tambahkan COMPOSER_MEMORY_LIMIT=-1 sebelum command

1.2.3   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
```

1.2.4   4. Create Database

Via MySQL Command:

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

1.2.3 3. Konfigurasi Lingkungan

 # Salin contoh .env
cp .env.example .env

 # Hasilkan kunci aplikasi
php artisan key:generate

Edit file .env:

```
# AplikasiAPP_NAME="Koperasi
Karyawan"
APP_ENV=local # produksi untuk server
liveAPP_DEBUG=true # false untuk
produksiAPP_URL=
http://localhost:8000APP_TIMEZONE=Asia/Jakarta

# Basis Data
DB_CONNECTION=mysqlDB_HOST=
127.0.0.1DB_PORT=
3306DB_DATABASE=
koperasiDB_USERNAME=
rootDB_PASSWORD=
your_password_here

# Mail（SMTP）
MAIL_MAILER=smtpMAIL_HOST=
smtp.gmail.comMAIL_PORT=
587MAIL_USERNAME=your_email@
gmail.comMAIL_PASSWORD=
your_app_passwordMAIL_ENCRYPTION=
tlsMAIL_FROM_ADDRESS=noreply@
koperasi.comMAIL_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_tokenWHATSAPP_SENDER=
6281234567890
```

1.2.4 4. Buat 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"

1.2.5 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

1.2.6 6. Storage Symlink

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

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

1.2.7 7. Install Frontend Dependencies

```
# Install NPM packages
npm install

# Atau dengan Yarn
yarn install
```

1.2.8 8. Compile Assets

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

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

1.2.9 9. Cache Configuration (Production Only)

```
# Cache config
php artisan config:cache

# Cache routes
php artisan route:cache
```

CREATE DATABASE koperasi CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci;exit;

Via phpMyAdmin: 1. Buka phpMyAdmin (http://localhost/phpmyadmin) 2. Klik "New" disidebar
3. Nama database: koperasi 4. Collation: utf8mb4_unicode_ci 5. Klik "Create"

1.2.5 5. Jalankan Migrations & Seeders

```
# Instalasi baru (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 - Anggota contoh - Kategori produk - Produk contoh
- Chart of accounts

1.2.6 6. Symlink Penyimpanan

```
# Buat tautan simbolik
php artisan storage:link
```

**Pengguna Windows: Jika error " symlink(): Protocol error", jalankan CMD/PowerShell
sebagai Administrator**

1.2.7 7. Instal Ketergantungan Frontend

```
# Instal paket NPM
npm install
```

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

1.2.8 8. Kompilasi Aset

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

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

1.2.9 9. Konfigurasi Cache (Hanya Produksi)

```
# Konfigurasi cache
php artisan config:cache
```

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

```
# Cache views
php artisan view:cache

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

1.2.10   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
```

1.2.11   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
```

1.3   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!

1.4   Additional Setup

1.4.1   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
```

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

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

1.2.10 10. Atur Izin (**Linux**/**Mac**)

 # Laravel penyimpanan & cache dapat ditulis
```
sudo chmod -R 775 storage bootstrap/cache
```

 # Atur kepemilikan (ganti 'www-data' sesuai pengguna web server)
```
sudo chown -R www-data:www-data storage bootstrap/cache
```

1.2.11 11. Mulai Server Pengembangan

 # Server bawaan Laravel
```
php artisan serve
```

 # Akses di: http://localhost:8000

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

1.3 Kredensial Login Default

Akun Admin: - Email: admin@koperasi.local - Password: passwordAkun

Pengurus: - Email: pengurus@koperasi.local - Password: passwordAkun

Member: - Email: member@koperasi.local - Password: passwordfi PENTING:

Ganti password default setelah login!

1.4 Pengaturan Tambahan

1.4.1 Pekerja Antrian (**Pekerjaan Latar Belakang**)

 # Mulai pekerja antrian
```
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
```

1.4.2 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.

1.4.3 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
```

1.5 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
```

```
autorestart=true
user=www-data
numprocs=2
redirect_stderr=true
stdout_logfile=/var/www/koperasi/storage/logs/worker.log
```

1.4.2 Penjadwal Tugas (Cron Jobs)

Linux/Mac:

```
# Edit crontab
crontab -e
```

```
# Tambahkan baris ini:
```
* * * * * cd /path-to-your-project && php artisan schedule:run >> /dev/null 2>&1 Windows:
Gunakan Task Scheduler untuk menjalankan perintah setiap menit.

1.4.3 Cache Redis (Opsional)

```
# Instal server Redis
```

```
# Instal ekstensi PHP Redis
pecl install redis
```

```
# Perbarui .env
CACHE_DRIVER=
redisSESSION_DRIVER=
redisQUEUE_CONNECTION=
redis

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

1.5 Daftar Periksa Verifikasi

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)

Perintah Uji:

```
# Kirim email uji
php artisan tinker
```

6

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

1.6 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
```

1.7 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

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

1.6 Pemecahan Masalah

Lihat TROUBLESHOOTING.md untuk solusi masalah umum.

Perbaikan Cepat:

```
# Hapus semua cache
php artisan cache:clearphp
artisan config:clearphp
artisan route:clearphp
artisan view:clear

# Buat ulang file autoload
composer dump-autoload

# Perbaiki izin penyimpanan
chmod -R 775 storage bootstrap/cache
```

1.7 Langkah Selanjutnya

1. fi Ganti kata sandi default
2. fi Siapkan email SMTP
3. fi Konfigurasi payment gateway
4. fi Siapkan jadwal cadangan
5. fi Kustomisasi merek (logo, nama koperasi)
6. fi Impor data anggota
7. fi Uji semua fitur
8. fi Deploy ke server produksi

Butuh Bantuan? - fi Email: support@kopkarskf.com - fi Dokumen Lengkap: FEATURES.md