

Contents

1	MAINTENANCE GUIDE	2
1.1	Document Information	2
1.2	Overview	2
1.3	MAINTENANCE SCHEDULE	2
1.3.1	Daily Tasks (Setiap Hari)	2
1.3.2	Weekly Tasks (Setiap Minggu)	2
1.3.3	Monthly Tasks (Setiap Bulan)	3
1.4	1. SYSTEM MONITORING	3
1.4.1	1.1 Server Health Check	3
1.4.2	1.2 Application Health	4
1.4.3	1.3 Log Monitoring	4
1.5	2. DATABASE MAINTENANCE	5
1.5.1	2.1 Daily Backup Verification	5
1.5.2	2.2 Database Performance Monitoring	5
1.5.3	2.3 Database Cleanup (Monthly)	6
1.6	3. CACHE MANAGEMENT	7
1.6.1	3.1 Clear Application Cache	7
1.6.2	3.2 Redis Cache (If Enabled)	7
1.7	4. FILE SYSTEM MAINTENANCE	8
1.7.1	4.1 Disk Space Management	8
1.7.2	4.2 File Permission Check	9
1.8	5. PERFORMANCE OPTIMIZATION	9
1.8.1	5.1 PHP-FPM Tuning	9
1.8.2	5.2 Nginx Optimization	9
1.8.3	5.3 Database Indexing	10
1.9	6. SECURITY MAINTENANCE	10
1.9.1	6.1 SSL Certificate Renewal	10
1.9.2	6.2 Security Updates	11
1.9.3	6.3 Audit Log Review	12
1.10	7. QUEUE & JOBS MANAGEMENT	12
1.10.1	7.1 Queue Worker Status	12
1.10.2	7.2 Failed Jobs Management	12
1.11	8. MONITORING & ALERTS	13
1.11.1	8.1 Application Metrics	13
1.11.2	8.2 Uptime Monitoring	13
1.11.3	8.3 Alert Configuration	14
1.12	9. TROUBLESHOOTING GUIDE	14
1.12.1	9.1 Application Not Accessible	14
1.12.2	9.2 Slow Performance	14
1.12.3	9.3 Database Connection Error	15
1.13	10. CHECKLISTS	16
1.13.1	10.1 Daily Checklist	16
1.13.2	10.2 Weekly Checklist	16
1.13.3	10.3 Monthly Checklist	17
1.14	11. ESCALATION CONTACTS	17
1.15	12. REFERENCES	17

Daftar Isi

1 fi PETUNJUK PERAWATAN 2

1.1 fi Informasi Dokumen	2
1.2 fi Ikhtisar	2
1.3 fi JADWAL PERAWATAN	2
1.3.1 Tugas Harian (Setiap Hari)	2
1.3.2 Tugas Mingguan (Setiap Minggu)	2
1.3.3 Tugas Bulanan (Setiap Bulan)	3
1.4 fi 1. PEMANTAUAN SISTEM	3
1.4.1 1.1 Pemeriksaan Kesehatan Server	3
1.4.2 1.2 Kesehatan Aplikasi	4
1.4.3 1.3 Pemantauan Log	4
1.5 fi 2. PEMELIHARAAN DATABASE	5
1.5.1 2.1 Verifikasi Cadangan Harian	5
1.5.2 2.2 Pemantauan Kinerja Database	5
1.5.3 2.3 Pembersihan Database (Bulanan)	6
1.6 fi 3. MANAJEMEN CACHE	7
1.6.1 3.1 Hapus Cache Aplikasi	7
1.6.2 3.2 Cache Redis (Jika Diaktifkan)	7
1.7 fi 4. PEMELIHARAAN SISTEM FILE	8
1.7.1 4.1 Manajemen Ruang Disk	8
1.7.2 4.2 Pemeriksaan Izin File	9
1.8 fi 5. OPTIMISASI KINERJA	9
1.8.1 5.1 Penyetelan PHP-FPM	9
1.8.2 5.2 Optimisasi Nginx	9
1.8.3 5.3 Pengindeksan Database	10
1.9 fi 6. PEMELIHARAAN KEAMANAN	10
1.9.1 6.1 Pembaruan Sertifikat SSL	10
1.9.2 6.2 Pembaruan Keamanan	11
1.9.3 6.3 Tinjauan Log Audit	12
1.10 fi 7. MANAJEMEN ANTRIAN & PEKERJAAN	12
1.10.1 7.1 Status Pekerja Antrian	12
1.10.2 7.2 Manajemen Pekerjaan Gagal	12
1.11 fi 8. MONITORING & PERINGATAN	13
1.11.1 8.1 Metrik Aplikasi	13
1.11.2 8.2 Monitoring Waktu Aktif	13
1.11.3 8.3 Konfigurasi Peringatan	14
1.12 fi 9. PANDUAN PEMECAHAN MASALAH	14
1.12.1 9.1 Aplikasi Tidak Dapat Diakses	14
1.12.2 9.2 Kinerja Lambat	14
1.12.3 9.3 Kesalahan Koneksi Database	15
1.13 fi 10. DAFTAR PERIKSA	16
1.13.1 10.1 Daftar Periksa Harian	16
1.13.2 10.2 Daftar Periksa Mingguan	16
1.13.3 10.3 Daftar Periksa Bulanan	17
1.14 fi 11. KONTAK ESKALASI	17
1.15 fi 12. REFERENSI	17

1 MAINTENANCE GUIDE

1.1 Document Information

Application: Koperasi Karyawan SKF

Version: 2.0.0

Maintainer: IT Team

Last Updated: 17 January 2026

1.2 Overview

Dokumen ini berisi panduan lengkap untuk **maintenance & monitoring** aplikasi Koperasi Karyawan SKF dalam operasional harian, mingguan, dan bulanan.

Target Audience: - System Administrator - DevOps Engineer - IT Support Team

1.3 MAINTENANCE SCHEDULE

1.3.1 Daily Tasks (Setiap Hari)

Time	Task	Duration	Priority
08:00	Check system uptime & accessibility	5 min	P0
08:10	Review overnight backup status	10 min	P0
08:30	Check error logs for critical issues	15 min	P0
09:00	Monitor disk space usage	5 min	P1
17:00	Review day's audit logs	10 min	P1
17:30	Check pending notifications queue	5 min	P2

Total Daily: ~50 minutes

1.3.2 Weekly Tasks (Setiap Minggu)

Day	Task	Duration
Monday	Review security logs & access patterns	30 min
Tuesday	Database performance analysis	45 min
Wednesday	Update system dependencies (check)	20 min
Thursday	Backup verification (test restore)	60 min
Friday	Weekly status report	30 min
Saturday	System health check comprehensive	45 min
Sunday	Full system backup (automated + verify)	15 min

Total Weekly: ~4 hours

1 PANDUAN PERAWATAN

1.1 Informasi Dokumen

Aplikasi: Koperasi Karyawan SKF
Versi: 2.0.0
Pemelihara: Tim
IT Terakhir Diperbarui: 17 Januari 2026

1.2 Ikhtisar

Dokumen ini berisi panduan lengkap untuk perawatan & pemantauan aplikasi Koperasi Karyawan SKF dalam operasional harian, mingguan, dan bulanan.

Target Audiens: - Administrator Sistem - Insinyur DevOps - Tim Dukungan IT

1.3 JADWAL PERAWATAN

1.3.1 Tugas Harian (Setiap Hari)

Waktu Tugas		Durasi Prioritas	
08:00	Periksa waktu aktif sistem & aksesibilitas	08:00 Periksa waktu aktif sistem & aksesibilitas 5 menit	P0
08:10	Tinjau status cadangan	10 menit	10 min P0
08:30	semalamanPeriksa log kesalahan untuk masalah kritis	08:30 Periksa log kesalahan untuk masalah kritis 15 menit P0	P1
09:00	Pantau penggunaan ruang disk	5 min	10 min P1
17:00	Tinjau log audit hari ini	10 min	P2
17:30	Periksa antrean notifikasi yang tertunda	17:30 Periksa antrean notifikasi yang tertunda 5 min	

Total Harian: ~50 menit

1.3.2 Tugas Mingguan (Setiap Minggu)

Hari	Tugas	Durasi
Senin	Tinjau log keamanan & pola akses	Senin Tinjau log keamanan & pola akses 30 menit
Selasa	Analisis kinerja basis data	45 menit
Rabu	Perbarui ketergantungan sistem (periksa)	20 menit
Kamis	Verifikasi cadangan (uji pemulihan)	60 menit
Jumat	Laporan status mingguan	30 menit
Sabtu	Pemeriksaan kesehatan sistem komprehensif	45 menit
Minggu	Cadangan sistem penuh (otomatis + verifikasi)	15 menit

Total Mingguan: ~4 jam

1.3.3 Monthly Tasks (Setiap Bulan)

Task	Schedule	Duration
Security patch review & apply	Week 1	2 hours
Database optimization & cleanup	Week 2	1.5 hours
SSL certificate renewal check	Week 2	15 min
Audit log archive	Last day of month	30 min
Performance report generation	Last day of month	1 hour
Disaster recovery drill	Last Sunday	2 hours

Total Monthly: ~7 hours

1.4 1. SYSTEM MONITORING

1.4.1 1.1 Server Health Check

Command:

```
# Check system resources
cd /var/www/koperasi
./scripts/health-check.sh
```

Manual Checks:

```
# CPU & Memory
htop
free -h

# Disk space (Alert if > 80%)
df -h

# Running services
systemctl status nginx
systemctl status php8.2-fpm
systemctl status mysql
systemctl status redis-server
systemctl status supervisor

# Network connectivity
ping -c 3 google.com
```

Expected Output:

```
CPU usage: < 70%
Memory usage: < 80%
Disk usage: < 80%
All services: active (running)
Network: accessible
```

1.3.3 Tugas Bulanan (Setiap Bulan)

Tugas	Jadwal	Durasi
Tinjau & terapkan patch keamanan	Tinjau & terapkan patch keamanan Minggu 1	2 jam
Optimasi & pembersihan database	Minggu 2	1.5 jam
Pemeriksaan perpanjangan sertifikat SSL	Minggu 2	15 menit
Arsip log auditPembuatan laporan kinerja	Hari terakhir bulan	30 menit
Latihan pemulihan bencana	Hari terakhir bulan	1 jam
	Minggu terakhir	2 jam

Total Bulanan: ~7 jam

1.4 1. MONITORING SISTEM

1.4.1 1.1 Pemeriksaan Kesehatan Server

Perintah:

Periksa sumber daya sistem

```
cd /var/www/koperasi
./scripts/health-check.sh
```

Pemeriksaan Manual:

CPU & Memori

```
htop
free -h
```

Ruang disk (Peringatan jika > 80%)

```
df -h
```

Layanan yang berjalan

```
systemctl status nginxsystemctl
status php8.2-fpmstatus systemctl
status mysqlsystemctl status
redis-serversystemctl status
supervisor
```

Konektivitas jaringan

```
ping -c 3 google.com
```

Output yang Diharapkan:

Penggunaan CPU: < 70%
Penggunaan Memori: < 80%
Penggunaan Disk: < 80%
Semua layanan: aktif (berjalan)
Jaringan: dapat diakses

If Issues: - CPU > 80%: Check PHP-FPM pool, identify slow queries - Memory > 90%: Restart PHP-FPM, clear cache - Disk > 85%: Clean old logs, backups

1.4.2 1.2 Application Health

URL Health Check:

```
# Via curl
curl -I https://kopkarskf.com
```

```
# Expected: HTTP/2 200
```

Laravel Health:

```
cd /var/www/koperasi

# Check application status
php artisan inspire # Should return inspirational quote

# Check database connection
php artisan tinker
>>> DB::connection()->getPdo(); exit;

# Check queue workers
php artisan queue:work --once
```

1.4.3 1.3 Log Monitoring

Location:

```
/var/www/koperasi/storage/logs/
  laravel.log          # Application logs
  audit.log            # Audit trail
  nginx-access.log     # Web server access
  nginx-error.log      # Web server errors
  php-fpm.log          # PHP errors
```

Daily Check:

```
# Check for errors in last 24 hours
cd /var/www/koperasi/storage/logs

# Critical errors
grep -i "error\|exception\|fatal" laravel.log | tail -50

# Failed logins
grep "Failed login" audit.log | tail -20
```

Jika Masalah: - CPU > 80%: Periksa kolam PHP-FPM, identifikasi kueri lambat - Memori > 90%: Restart PHP-FPM, bersihkan cache - Disk > 85%: Bersihkan log lama, cadangan

1.4.2 1.2 Kesehatan Aplikasi

Pemeriksaan Kesehatan URL:

Via curl

`curl -I https://kopkarskf.com`

Diharapkan: HTTP/2 200

Kesehatan Laravel:

`cd /var/www/koperasi`

Periksa status aplikasi

php artisan inspire # Harus mengembalikan kutipan inspiratif

Periksa koneksi database

php artisan tinker>>> DB::connection

(->getPdo()); exit;

Periksa pekerja antrean

php artisan queue:work --once

1.4.3 1.3 Pemantauan Log

Lokasi:

<code>/var/www/koperasi/storage/logs/</code>	
<code>laravel.log</code>	<code># Log aplikasi# Jejak</code>
<code>audit.log</code>	<code>audit# Akses server</code>
<code>nginx-access.log</code>	<code>web# Kesalahan</code>
<code>nginx-error.log</code>	<code>server web#</code>
<code>php-fpm.log</code>	<code>Kesalahan PHP</code>

Pemeriksaan Harian:

Periksa kesalahan dalam 24 jam terakhir

`cd /var/www/koperasi/storage/logs`

Kesalahan kritis

grep -i "error\|exception\|fatal" laravel.log | tail -50

Login gagal

grep "Failed login" audit.log | tail -20

```
# 500 errors
```

```
grep " 500 " nginx-access.log | tail -20
```

Alert Triggers: - > 10 errors/hour in laravel.log - > 5 failed logins from same IP - > 5 HTTP 500 errors/hour

Action: - Investigate error context - Check stack trace - Apply fix - Monitor for recurrence

1.5 2. DATABASE MAINTENANCE

1.5.1 2.1 Daily Backup Verification

Location:

/var/www/koperasi/storage/app/backups/

Format: koperasi_YYYYMMDD_HHMMSS.sql.gz

Check:

```
# Verify backup exists (created at 02:00 daily)
```

```
ls -lh /var/www/koperasi/storage/app/backups/ | grep $(date +%Y%m%d)
```

```
# Check file size (should be > 10MB)
```

```
du -h /var/www/koperasi/storage/app/backups/koperasi_$(date +%Y%m%d)*.sql.gz
```

```
# Test backup integrity
```

```
gunzip -t /var/www/koperasi/storage/app/backups/koperasi_$(date +%Y%m%d)*.sql.gz
```

```
echo $? # Should output: 0 (success)
```

If Backup Missing:

```
# Manual backup
```

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

```
php artisan backup:run
```

1.5.2 2.2 Database Performance Monitoring

Weekly Check:

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

```
-- Check database size
```

```
SELECT
```

```
    table_schema AS 'Database',
```

```
    ROUND(SUM(data_length + index_length) / 1024 / 1024, 2) AS 'Size (MB)'
```

```
FROM information_schema.tables
```

```
WHERE table_schema = 'koperasi'
```

```
GROUP BY table_schema;
```

```
-- Check largest tables
```

500 kesalahan

grep " 500 " nginx-access.log | tail -20

Pemicu Peringatan: - > 10 kesalahan/jam di laravel.log - > 5 login gagal dari IP yang sama - > 5 kesalahan HTTP500/jam

Tindakan: - Selidiki konteks kesalahan - Periksa jejak tumpukan - Terapkan perbaikan - Pantau untuk kejadian ulang

1.5 2. PEMELIHARAAN DATABASE

1.5.1 2.1 Verifikasi Cadangan Harian

Lokasi:

/var/www/koperasi/storage/app/backups/

Format: koperasi_YYYYMMDD_HHMMSS.sql.gz

Periksa:

Verifikasi cadangan ada (dibuat pada 02:00 setiap hari)

ls -lh /var/www/koperasi/storage/app/backups/ | grep \$(date +%Y%m%d)

Periksa ukuran file (seharusnya > 10MB)

du -h /var/www/koperasi/storage/app/backups/koperasi_\$(date +%Y%m%d).sql.gz*

Uji integritas cadangan

gunzip -t /var/www/koperasi/storage/app/backups/koperasi_\$(date +%Y%m%d).sql.gz*
echo \$? # Harus mengeluarkan: 0 (sukses)

Jika Cadangan Hilang:

Cadangan manual

cd /var/www/koperasiphp

artisan backup:run

1.5.2 2.2 Pemantauan Kinerja Database

Pemeriksaan Mingguan:

mysql -u root -p koperasi

-- Periksa ukuran

databaseSELECT

table_schema AS 'Database',

ROUND(SUM(data_length + index_length) / 1024 / 1024, 2) AS 'Ukuran (MB)'

FROM

information_schema.tablesWHERE

table_schema = 'koperasi'GROUP BY

table_schema;

-- Periksa tabel terbesar

```

SELECT
    table_name AS 'Table',
    ROUND(((data_length + index_length) / 1024 / 1024), 2) AS 'Size (MB)'
FROM information_schema.tables
WHERE table_schema = 'koperasi'
ORDER BY (data_length + index_length) DESC
LIMIT 10;

-- Check slow queries (> 2 seconds)
SELECT * FROM mysql.slow_log
WHERE sql_text NOT LIKE '%information_schema%'
ORDER BY query_time DESC
LIMIT 10;

-- Check table fragmentation
SELECT
    table_name,
    ROUND(data_length / 1024 / 1024, 2) AS data_mb,
    ROUND(data_free / 1024 / 1024, 2) AS free_mb,
    ROUND((data_free / data_length) * 100, 2) AS fragmentation_pct
FROM information_schema.tables
WHERE table_schema = 'koperasi'
    AND data_free > 0
ORDER BY fragmentation_pct DESC;

exit;

If Fragmentation > 10%:

mysql -u root -p koperasi

-- Optimize table
OPTIMIZE TABLE table_name;

-- Or optimize all tables
mysqlcheck -u root -p --optimize koperasi

```

1.5.3 2.3 Database Cleanup (Monthly)

Old Data Cleanup:

```

mysql -u root -p koperasi

-- Delete old sessions (> 7 days)
DELETE FROM sessions
WHERE last_activity < UNIX_TIMESTAMP(DATE_SUB(NOW(), INTERVAL 7 DAY));

-- Delete old failed_jobs (> 30 days)

```

```

SELECT
    table_name AS 'Tabel',
    ROUND(((data_length + index_length) / 1024 / 1024), 2) AS 'Ukuran (MB)'
FROM information_schema.tables
WHERE table_schema = 'koperasi'
ORDER BY (data_length + index_length) DESC
LIMIT 10;

```

```

-- Periksa kueri lambat (> 2 detik)
SELECT * FROM mysql.slow_log
WHERE sql_text NOT LIKE '%information_schema%'
ORDER BY query_time
DESC LIMIT 10;

```

```

-- Periksa fragmentasi
tabelSELECT
    table_name,
    ROUND(data_length / 1024 / 1024, 2) AS data_mb,
    ROUND(data_free / 1024 / 1024, 2) AS free_mb,
    ROUND((data_free / data_length) * 100, 2) AS fragmentation_pct
FROM information_schema.tables
WHERE table_schema = 'koperasi'
    AND data_free > 0
ORDER BY fragmentation_pct DESC;

exit;

```

Jika Fragmentasi > 10%:

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

```

-- Optimalkan tabelOPTIMIZE
TABLE table_name;

```

```

-- Atau optimalkan semua tabel
mysqlcheck -u root -p --optimize koperasi

```

1.5.3 2.3 Pembersihan Database (Bulanan)

Pembersihan Data Lama:

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

```

-- Hapus sesi lama (> 7 hari)
DELETE FROM sessions
DIMANA last_activity < UNIX_TIMESTAMP(DATE_SUB(NOW(), INTERVAL 7
HARI)); -- Hapus failed_jobs lama (> 30 hari)

```

```

DELETE FROM failed_jobs
WHERE failed_at < DATE_SUB(NOW(), INTERVAL 30 DAY);

-- Delete old password_reset_tokens (> 1 day)
DELETE FROM password_reset_tokens
WHERE created_at < DATE_SUB(NOW(), INTERVAL 1 DAY);

-- Archive old audit_logs (> 1 year to separate table)
INSERT INTO audit_logs_archive
SELECT * FROM audit_logs
WHERE created_at < DATE_SUB(NOW(), INTERVAL 1 YEAR);

DELETE FROM audit_logs
WHERE created_at < DATE_SUB(NOW(), INTERVAL 1 YEAR);

exit;

```

1.6 3. CACHE MANAGEMENT

1.6.1 3.1 Clear Application Cache

When to Clear: - After code deployment - After configuration changes - Performance degradation
 - Weekly maintenance

Commands:

```

cd /var/www/koperasi

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

# Rebuild caches (production only)
php artisan config:cache
php artisan route:cache
php artisan view:cache

# Restart services
sudo systemctl restart php8.2-fpm
sudo systemctl reload nginx

```

1.6.2 3.2 Redis Cache (If Enabled)

Monitor:

```
DELETE FROM failed_jobs
WHERE failed_at < DATE_SUB(NOW(), INTERVAL 30 DAY);
```

```
-- Hapus password_reset_tokens lama (> 1 hari)
DELETE FROM password_reset_tokens
WHERE created_at < DATE_SUB(NOW(), INTERVAL 1 DAY);
```

```
-- Arsipkan audit_logs lama (> 1 tahun ke tabel terpisah)
INSERT INTO audit_logs_archive
SELECT * FROM audit_logs
WHERE created_at < DATE_SUB(NOW(), INTERVAL 1
```

```
YEAR); DELETE FROM audit_logs
WHERE created_at < DATE_SUB(NOW(), INTERVAL 1
YEAR); exit;
```

1.6 3. MANAJEMEN CACHE

1.6.1 3.1 Hapus Cache Aplikasi

Kapan untuk Menghapus: - Setelah penerapan kode - Setelah perubahan konfigurasi - Penurunan kinerja - Pemeliharaan mingguan

Perintah:

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

```
# Hapus semua cache
```

```
php artisan cache:clearphp
artisan config:clearphp
artisan route:clearphp
artisan view:clear
```

```
# Bangun kembali cache (hanya produksi)
```

```
php artisan config:cachephp
artisan route:cachephp
artisan view:cache
```

```
# Mulai ulang layanan
```

```
sudo systemctl restart php8.2-
fpm sudo systemctl reload nginx
```

1.6.2 3.2 Redis Cache (Jika Diaktifkan)

Monitor:

```
redis-cli

# Check memory usage
INFO memory

# Check key count
DBSIZE

# Check slow commands
SLOWLOG GET 10

exit
```

Clear Redis:

```
redis-cli FLUSHALL
```

1.7 4. FILE SYSTEM MAINTENANCE

1.7.1 4.1 Disk Space Management

Monitor:

```
# Overall disk usage
df -h

# Find large directories
du -sh /var/www/koperasi/* | sort -hr | head -10

# Find large files
find /var/www/koperasi -type f -size +100M -exec ls -lh {} \;
```

Cleanup Actions:

1. Old Log Files (> 30 days):

```
find /var/www/koperasi/storage/logs -name "*.log" -mtime +30 -delete
```

2. Old Backup Files (> 90 days):

```
find /var/www/koperasi/storage/app/backups -name "*.sql.gz" -mtime +90 -delete
```

3. Temporary Files:

```
# Session files
find /var/www/koperasi/storage/framework/sessions -type f -mtime +7 -delete
```

```
# Cache files
find /var/www/koperasi/storage/framework/cache -type f -mtime +7 -delete
```

```
# Temporary uploads
find /var/www/koperasi/storage/app/tmp -type f -mtime +1 -delete
```

redis-cli

Periksa penggunaan memori

INFO memory

Periksa jumlah kunci

DBSIZE

Periksa perintah lambat

SLOWLOG GET 10

exit

Bersihkan Redis:

redis-cli FLUSHALL

1.7 4. PEMELIHARAAN SISTEM FILE

1.7.1 4.1 Manajemen Ruang Disk

Monitor:

Penggunaan disk secara keseluruhan

df -h

Temukan direktori besar

du -sh /var/www/koperasi/* | sort -hr | head -10

Temukan file besar

find /var/www/koperasi -type f -size +100M -exec ls -lh {} \;

Tindakan Pembersihan:

1. File Log Lama (> 30 hari):

find /var/www/koperasi/storage/logs -name "*.log" -mtime +30 -delete

2. File Cadangan Lama (> 90 hari):

find /var/www/koperasi/storage/app/backups -name "*.sql.gz" -mtime +90 -delete

3. File Sementara:

File sesi

find /var/www/koperasi/storage/framework/sessions -type f -mtime +7 -delete

File cache

find /var/www/koperasi/storage/framework/cache -type f -mtime +7 -delete

Unggahan sementara

find /var/www/koperasi/storage/app/tmp -type f -mtime +1 -delete

4. Compress Old Logs:

```
cd /var/www/koperasi/storage/logs
gzip *.log.$(date -d "7 days ago" +%Y-%m-%d)
```

1.7.2 4.2 File Permission Check

Verify Permissions:

```
cd /var/www/koperasi

# Check ownership
ls -la storage bootstrap/cache

# Expected:
# drwxrwxr-x www-data www-data storage
# drwxrwxr-x www-data www-data bootstrap/cache

# Fix if needed
sudo chown -R www-data:www-data storage bootstrap/cache
sudo chmod -R 775 storage bootstrap/cache
```

1.8 5. PERFORMANCE OPTIMIZATION

1.8.1 5.1 PHP-FPM Tuning

Check Current Settings:

```
sudo cat /etc/php/8.2/fpm/pool.d/www.conf | grep -E "pm.max_children|pm.start_servers|pm.min_s"
```

Recommended Settings (4GB RAM server):

```
pm = dynamic
pm.max_children = 50
pm.start_servers = 10
pm.min_spare_servers = 5
pm.max_spare_servers = 20
pm.max_requests = 500
```

Apply Changes:

```
sudo systemctl restart php8.2-fpm
```

1.8.2 5.2 Nginx Optimization

Enable Gzip:

```
# /etc/nginx/nginx.conf
gzip on;
```

4. Kompres Log Lama:

```
cd /var/www/koperasi/storage/logs
gzip *.log.$(date -d "7 hari yang lalu" +%Y-%m-%d)
```

1.7.2 4.2 Cek Izin File

Verifikasi Izin:

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

Cek kepemilikan

```
ls -la storage bootstrap/cache
```

Diharapkan :

drwxrwxr-x www-data www-data storage

drwxrwxr-x www-data www-data bootstrap/cache

Perbaiki jika diperlukan

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

```
bootstrap/cachesudo chmod -R 775 storage bootstrap/cache
```

1.8 5. OPTIMISASI KINERJA

1.8.1 5.1 Penyetelan PHP-FPM

Periksa Pengaturan Saat Ini:

```
sudo cat /etc/php/8.2/fpm/pool.d/www.conf | grep -E "pm.max_children|pm.start_servers|pm.min_spPen  
RAM 4GB):
```

pm = dinamis

pm.max_children = 50

pm.start_servers = 10

pm.min_spare_servers = 5

pm.max_spare_servers = 20

pm.max_requests = 500

Terapkan Perubahan:

```
sudo systemctl restart php8.2-fpm
```

1.8.2 5.2 Optimasi Nginx

Aktifkan Gzip:

```
# /etc/nginx/nginx.conf
```

```
gzip on;
```

```
gzip_types text/plain text/css application/json application/javascript text/xml application/xml
gzip_min_length 1000;
```

Browser Cache:

```
# Static assets
location ~* \.(jpg|jpeg|png|gif|ico|css|js|svg|woff|woff2)$ {
    expires 365d;
    add_header Cache-Control "public, immutable";
}
```

Reload Nginx:

```
sudo nginx -t
sudo systemctl reload nginx
```

1.8.3 5.3 Database Indexing

Check Missing Indexes:

```
-- Find queries without indexes
SELECT * FROM mysql.slow_log
WHERE sql_text NOT LIKE '%FROM information_schema%'
ORDER BY query_time DESC
LIMIT 20;
```

Add Indexes (Example):

```
-- If you see frequently queried columns without index
CREATE INDEX idx_members_status ON members(status);
CREATE INDEX idx_loans_status ON loans(status, approved_at);
CREATE INDEX idx_transactions_date ON transactions(created_at);
```

1.9 6. SECURITY MAINTENANCE

1.9.1 6.1 SSL Certificate Renewal

Check Expiry:

```
# Check certificate expiration
sudo certbot certificates
```

```
# Expected output shows expiry date
```

Auto-Renewal (Let's Encrypt):

```
# Test renewal
sudo certbot renew --dry-run
```

```
# Cron job should be in place
```

```
gzip_types text/plain text/css application/json application/javascript text/xml application/xml
gzip_min_length 1000;
```

Cache Browser:

```
# Aset statis
location ~* \. (jpg|jpeg|png|gif|ico|css|js|svg|woff|woff2)$ {
    expires 365d;
    add_header Cache-Control "public, immutable";
}
```

Muat ulang Nginx:

```
sudo nginx -t
sudo systemctl reload nginx
```

1.8.3 5.3 Pengindeksan Basis Data

Periksa Indeks yang Hilang:

```
-- Temukan kueri tanpa indeks
SELECT * FROM mysql.slow_log
WHERE sql_text NOT LIKE '%FROM information_schema%'
ORDER BY query_time
DESCLIMIT 20;
```

Tambahkan Indeks (Contoh):

```
-- Jika Anda melihat kolom yang sering ditanyakan tanpa indeks
CREATE INDEX idx_members_status ON members(status);
CREATE INDEX idx_loans_status ON loans(status, approved_at);
CREATE INDEX idx_transactions_date ON transactions(created_at);
```

1.9 6. PEMELIHARAAN KEAMANAN

1.9.1 6.1 PERPANJANGAN SERTIFIKAT SSL

Periksa Masa Berlaku:

```
# Periksa masa berlaku sertifikat
sudo certbot certificates
```

```
# Output yang diharapkan menunjukkan tanggal kedaluwarsa
```

Perpanjangan Otomatis (Let's Encrypt):

```
# Uji perpanjangan
sudo certbot renew --dry-run
```

```
# Tugas cron harus ada
```

```
cat /etc/cron.d/certbot
# Expected: 0 */12 * * * root certbot renew --quiet
```

Manual Renewal (if needed):

```
sudo certbot renew
sudo systemctl reload nginx
```

1.9.2 6.2 Security Updates

Check for Updates:

```
# System packages
sudo apt update
sudo apt list --upgradable
```

```
# Composer dependencies
cd /var/www/koperasi
composer outdated
```

```
# NPM dependencies
npm outdated
```

Apply Updates (Carefully):

```
# Security updates only
sudo apt upgrade -y

# Composer (test in staging first!)
composer update --with-dependencies
```

```
# NPM (test in staging first!)
npm update
npm run build
```

After Updates:

```
# Clear cache
php artisan cache:clear
php artisan config:clear

# Restart services
sudo systemctl restart php8.2-fpm
sudo systemctl reload nginx

# Verify application
curl -I https://kopkarskf.com
```

```
cat /etc/cron.d/certbot  
# Diharapkan: 0 */12 * * * root certbot renew --quiet
```

Pembaruan Manual (jika diperlukan):

```
sudo certbot renew  
sudo systemctl reload nginx
```

1.9.2 6.2 Pembaruan Keamanan

Periksa Pembaruan:

```
# Paket sistem  
sudo apt update  
sudo apt list --upgradable
```

```
# Ketergantungan Composer  
cd  
/var/www/koperasicomposer  
outdated
```

```
# Ketergantungan NPM  
npm outdated
```

Terapkan Pembaruan (Dengan Hati-hati):

```
# Pembaruan keamanan saja  
sudo apt upgrade -y
```

```
# Composer (uji di staging terlebih dahulu!)  
composer update --with-dependencies
```

```
# NPM (uji di staging terlebih dahulu!)  
npm update  
run build
```

Setelah Pembaruan:

```
# Hapus cache  
php artisan cache:clear  
artisan config:clear
```

```
# Mulai ulang layanan  
sudo systemctl restart php8.2-  
fpm  
sudo systemctl reload nginx
```

```
# Verifikasi aplikasi  
curl -I https://kopkarskf.com
```

1.9.3 6.3 Audit Log Review

Weekly Review:

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

```
# Failed login attempts
```

```
php artisan tinker
```

```
>>> DB::table('audit_logs')->where('action', 'failed_login')->where('created_at', '>', now())->
```

```
# Unauthorized access (403)
```

```
>>> DB::table('audit_logs')->where('action', 'unauthorized')->where('created_at', '>', now())->
```

```
# Critical data changes
```

```
>>> DB::table('audit_logs')->whereIn('model', ['Loan', 'ShuDistribution', 'Member'])->where('c
```

```
exit;
```

1.10 7. QUEUE & JOBS MANAGEMENT

1.10.1 7.1 Queue Worker Status

Check:

```
# Supervisor status (queue worker)
```

```
sudo supervisorctl status
```

```
# Expected: laravel-worker:* RUNNING
```

```
# Check failed jobs
```

```
mysql -u root -p koperasi -e "SELECT COUNT(*) FROM failed_jobs;"
```

Restart Workers:

```
# Graceful restart
```

```
php artisan queue:restart
```

```
# Force restart via Supervisor
```

```
sudo supervisorctl restart laravel-worker:*
```

1.10.2 7.2 Failed Jobs Management

View Failed Jobs:

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

```
php artisan queue:failed
```

Retry Failed Jobs:

1.9.3 6.3 Tinjauan Log Audit

Tinjauan Mingguan:

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

```
# Upaya login yang gagal
```

```
php artisan tinker
```

```
>>> DB::table('audit_logs')->where('action', 'failed_login')->where('created_at', '>', now())->s
```

```
# Akses tidak sah (403)
```

```
>>> DB::table('audit_logs')->where('action', 'unauthorized')->where('created_at', '>', now())->s
```

```
# Perubahan data kritis
```

```
>>> DB::table('audit_logs')->whereIn('model', ['Loan', 'ShuDistribution', 'Member'])->where('cr
```

```
exit;
```

1.10 7. MANAJEMEN ANTRIAN & PEKERJAAN

1.10.1 7.1 Status Pekerja Antrian

Periksa:

```
# Status pengawas (pekerja antrian)
```

```
sudo supervisorctl status
```

```
# Diharapkan: laravel-worker: * BERJALAN
```

```
# Periksa pekerjaan yang gagal
```

```
mysql -u root -p koperasi -e "SELECT COUNT(*) FROM failed_jobs;"
```

Mulai Ulang Pekerja:

```
# Restart yang Anggun
```

```
php artisan queue:restart
```

```
# Paksa restart melalui Supervisor
```

```
sudo supervisorctl restart laravel-worker:*
```

1.10.2 7.2 Manajemen Pekerjaan Gagal

Lihat Pekerjaan Gagal:

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

```
artisan queue:failed
```

Coba Kembali Pekerjaan Gagal:

```
# Retry specific job
php artisan queue:retry JOB_ID
```

```
# Retry all
php artisan queue:retry all
```

Clear Old Failed Jobs:

```
# Flush all failed jobs (CAUTION!)
php artisan queue:flush
```

1.11 8. MONITORING & ALERTS

1.11.1 8.1 Application Metrics

Daily Metrics to Track:

```
# Via tinkler
php artisan tinkler
```

```
# Today's statistics
>>> $today = today();
>>> [
    'transactions' => DB::table('transactions')->whereDate('created_at', $today)->count(),
    'new_members' => DB::table('members')->whereDate('created_at', $today)->count(),
    'loan_applications' => DB::table('loans')->whereDate('created_at', $today)->count(),
    'errors' => File::lines(storage_path('logs/laravel.log'))->filter(fn($line) => str_contains(
]);

exit;
```

Weekly Report:

```
# Generate report
php artisan report:weekly
```

```
# Or manually create Excel report from database
```

1.11.2 8.2 Uptime Monitoring

External Monitoring (Recommended): - UptimeRobot (free): <https://uptimerobot.com> - Monitor: <https://kopkarskf.com> - Alert: Email + WhatsApp - Interval: 5 minutes - Alert if: Down for 2+ checks

Self-hosted Alternative:

```
# Simple cron job ping
crontab -e
```

Coba ulang pekerjaan tertentu

php artisan queue:retry JOB_ID

Coba ulang semua

php artisan queue:retry all

Hapus Pekerjaan Gagal Lama:

Kosongkan semua pekerjaan gagal

(PERHATIAN!) php artisan queue:flush

1.11 8. MONITORING & ALERTS

1.11.1 8.1 Metrik Aplikasi

Metrik Harian untuk Dilacak:

Melalui tinker

php artisan tinker

Statistik Hari Ini

>>> **\$today = today();**

>>> [

'transactions' => DB::table('transactions')->whereDate('created_at', \$today)->count(),
'new_members' => DB::table('members')->whereDate('created_at', \$today)->count(),
'loan_applications' => DB::table('loans')->whereDate('created_at', \$today)->count(),
'errors' => File::lines(storage_path('logs/laravel.log'))->filter(fn(\$line) => str_contains

];

exit;

Laporan Mingguan:

Buat laporan

php artisan report:weekly

Atau buat laporan Excel secara manual dari database

1.11.2 8.2 Pemantauan Waktu Aktif

Pemantauan Eksternal (Disarankan): - UptimeRobot (gratis): <https://uptimerobot.com> - Monitor: <https://kopkarskf.com> - Pemberitahuan: Email + WhatsApp - Interval: 5 menit - Pemberitahuan jika: Down selama 2+ pemeriksaan

Alternatif yang Di-host Sendiri:

Panggilan cron sederhana

crontab -e

```
# Add this line (every 5 minutes)
*/5 * * * * curl -f https://kopkarskf.com || echo "Site down!" | mail -s "ALERT: Site Down" admin@kopkarskf.com
```

1.11.3 8.3 Alert Configuration

Email Alerts: - Disk space > 85% - Backup failed - Critical errors > 10/hour - Database connection failed - SSL certificate expiring < 30 days

WhatsApp Alerts (Critical): - Application down > 5 minutes - Database crash - Security breach detected

1.12 9. TROUBLESHOOTING GUIDE

1.12.1 9.1 Application Not Accessible

Symptoms: Site returns 502 Bad Gateway or timeout

Checklist:

```
# 1. Check Nginx
sudo systemctl status nginx
sudo nginx -t
# If error, check config:
sudo nano /etc/nginx/sites-available/koperasi

# 2. Check PHP-FPM
sudo systemctl status php8.2-fpm
# If not running:
sudo systemctl start php8.2-fpm

# 3. Check PHP-FPM socket
ls -la /var/run/php/php8.2-fpm.sock
# Should exist with proper permissions

# 4. Check error logs
sudo tail -50 /var/log/nginx/error.log
sudo tail -50 /var/log/php8.2-fpm.log

# 5. Restart all
sudo systemctl restart php8.2-fpm
sudo systemctl reload nginx
```

1.12.2 9.2 Slow Performance

Symptoms: Pages loading > 5 seconds

Diagnosis:

Tambahkan baris ini (setiap 5 menit)

```
* /5 * * * * curl -f https://kopkarskf.com | | echo "Situs tidak dapat diakses!" | mail -s "PERINGATAN: Situs Tidak Dapat Diakses" adm
```

1.11.3 8.3 Konfigurasi Peringatan

Peringatan Email: - Ruang disk > 85% - Cadangan gagal - Kesalahan kritis > 10/jam - Koneksi database gagal - Sertifikat SSL akan kedaluwarsa < 30 hari

Peringatan WhatsApp (Kritis): - Aplikasi tidak berjalan > 5 menit - Kerusakan database - Pelanggaran keamanan terdeteksi

1.12 9. PANDUAN PEMECAHAN MASALAH

1.12.1 9.1 Aplikasi Tidak Dapat DiaksesGejala:

Situs mengembalikan 502 Bad Gateway atau

timeoutDaftar Periksa:

1. Periksa Nginx

sudo systemctl status

nginx**sudo nginx -t**

Jika ada kesalahan, periksa konfigurasi:

sudo nano /etc/nginx/sites-available/koperasi

2. Periksa PHP-FPM

sudo systemctl status php8.2-fpm

Jika tidak berjalan:

sudo systemctl start php8.2-fpm

3. Periksa soket PHP-FPM

ls -la /var/run/php/php8.2-fpm.sock

Harus ada dengan izin yang tepat

4. Periksa log kesalahan

sudo tail -50 /var/log/nginx/error.log

sudo tail -50 /var/log/php8.2-fpm.log

5. Restart semua

sudo systemctl restart php8.2-

fpm**sudo systemctl reload nginx**

1.12.2 9.2 Kinerja LambatGejala:

Halaman memuat > 5

detik**Diagnosis:**

```

# 1. Check server resources
htop
# Look for high CPU/memory usage

# 2. Check database
mysql -u root -p koperasi
SELECT * FROM mysql.slow_log ORDER BY query_time DESC LIMIT 10;

# 3. Check PHP-FPM pool
sudo tail -50 /var/log/php8.2-fpm.log
# Look for: "max_children reached"

# 4. Enable Laravel debug bar (temporarily)
php artisan debugbar:enable

# 5. Check for N+1 queries
# (View page with debugbar enabled)

```

Solutions:

```

# Increase PHP-FPM children
sudo nano /etc/php/8.2/fpm/pool.d/www.conf
# pm.max_children = 50 # Increase this

# Optimize database
mysql -u root -p
OPTIMIZE TABLE table_name;

# Clear cache
php artisan cache:clear
php artisan view:clear

# Restart
sudo systemctl restart php8.2-fpm

```

1.12.3 9.3 Database Connection Error

Symptoms: “SQLSTATE[HY000] [2002] Connection refused”

Fix:

```

# 1. Check MySQL service
sudo systemctl status mysql
# If not running:
sudo systemctl start mysql

# 2. Check MySQL credentials
php artisan config:show database

```

1. Periksa sumber daya server

htop

Cari penggunaan CPU/memori yang tinggi

2. Periksa database

mysql -u root -p koperasi

SELECT * FROM mysql.slow_log ORDER BY query_time DESC LIMIT 10;

3. Periksa kolam PHP-FPM

sudo tail -50 /var/log/php8.2-fpm.log

Cari: "max_children reached"

4. Aktifkan debug bar Laravel (sementara)

php artisan debugbar:enable

5. Periksa untuk kueri N+1

(Lihat halaman dengan debugbar diaktifkan)

Solusi:

Tingkatkan anak PHP-FPM

sudo nano /etc/php/8.2/fpm/pool.d/www.conf

pm.max_children = 50 # Tingkatkan ini

Optimalkan basis data

mysql -u root -p

OPTIMIZE TABLE table_name;

Hapus cache

php artisan cache:clear

artisan view:clear

Mulai ulang

sudo systemctl restart php8.2-fpm

1.12.3 9.3 Kesalahan Koneksi Database

Gejala: "SQLSTATE[HY000] [2002] Koneksi ditolak"

Perbaiki:

1. Periksa layanan MySQL

sudo systemctl status mysql

Jika tidak berjalan:

sudo systemctl start mysql

2. Periksa kredensial MySQL

php artisan config:show database

```

# 3. Test connection
mysql -h 127.0.0.1 -u DB_USERNAME -p
# Enter password from .env

# 4. Check MySQL logs
sudo tail -50 /var/log/mysql/error.log

# 5. Restart MySQL if needed
sudo systemctl restart mysql

```

1.13 10. CHECKLISTS

1.13.1 10.1 Daily Checklist

Date: ___/___/_____
 By: _____

Morning (08:00-09:00):
 Site accessible (curl test)
 Backup completed (check file)
 Error logs reviewed (< 10 errors)
 Disk space OK (< 80%)
 All services running

Evening (17:00-18:00):
 Audit logs reviewed
 Queue jobs processed
 No pending alerts

Notes: _____

1.13.2 10.2 Weekly Checklist

Week of: ___/___/_____
 By: _____

Security logs reviewed (no anomalies)
 Database performance check
 Dependency updates checked
 Backup restore tested
 Weekly report generated
 System health comprehensive

Issues Found: _____
 Actions Taken: _____

```
# 3. Test connection
mysql -h 127.0.0.1 -u DB_USERNAME -p
# Enter password from .env

# 4. Check MySQL logs
sudo tail -50 /var/log/mysql/error.log

# 5. Restart MySQL if needed
sudo systemctl restart mysql
```

1.13 10. CHECKLISTS

1.13.1 10.1 Daily Checklist

Date: ___/___/_____
By: _____

Morning (08:00-09:00):

- Site accessible (curl test)
- Backup completed (check file)
- Error logs reviewed (< 10 errors)
- Disk space OK (< 80%)
- All services running

Evening (17:00-18:00):

- Audit logs reviewed
- Queue jobs processed
- No pending alerts

Notes: _____

1.13.2 10.2 Weekly Checklist

Week of: ___/___/_____
By: _____

- Security logs reviewed (no anomalies)
- Database performance check
- Dependency updates checked
- Backup restore tested
- Weekly report generated
- System health comprehensive

Issues Found: _____
Actions Taken: _____

1.13.3 10.3 Monthly Checklist

Month: _____

By: _____

Security patches applied
Database optimized
SSL certificate valid (> 30 days)
Audit logs archived
Performance report generated
Disaster recovery drill passed

Summary: _____

1.14 11. ESCALATION CONTACTS

Issue	Contact	Phone	Email
Application down	DevOps Lead	+62-xxx	devops@kopkarskf.com
Database crash	DBA	+62-xxx	dba@kopkarskf.com
Security incident	Security Team	+62-xxx	security@kopkarskf.com
Critical bug	Dev Team	+62-xxx	dev@kopkarskf.com

Escalation Path: 1. IT Support (1st tier) - Basic troubleshooting 2. System Admin (2nd tier) - Server/database issues 3. DevOps Lead (3rd tier) - Complex infrastructure 4. CTO (4th tier) - Critical business impact

1.15 12. REFERENCES

- [Laravel Documentation](#)
 - [TROUBLESHOOTING.md](#)
 - [DEPLOYMENT.md](#)
 - [SECURITY.md](#)
-

Maintainer: IT Operations Team

Contact: ops@kopkarskf.com

Last Reviewed: 17 January 2026

Next Review: 17 April 2026 (Quarterly)

1.13.3 10.3 Daftar Periksa Bulanan

Bulan: _____

Oleh: _____

Tambalan keamanan diterapkan
Database dioptimalkan Sertifikat
SSL valid (> 30 hari) Log audit
diarsipkan Laporan kinerja
dihasilkan Latihan pemulihan
bencana berhasil

Ringkasan: _____

1.14 11. KONTAK ESKALASI

Masalah	Kontak	Telepon	Email
Aplikasi tidak berfungsi	Pemimpin DevOps	+62-xxx	devops@kopkarskf.com
Kecelakaan database	DBA	+62-xxx	dba@kopkarskf.com
Insiden keamanan	Tim Keamanan	+62-xxx	security@kopkarskf.com
Bug kritis	Tim Dev	+62-xxx	dev@kopkarskf.com

Jalur Eskalasi: 1. Dukungan TI (tingkat 1) - Pemecahan masalah dasar 2. Admin Sistem (tingkat 2) - Masalah server/basis data 3. Pemimpin DevOps (tingkat 3) - Infrastruktur kompleks 4. CTO (tingkat 4) - Dampak bisnis kritis

1.15 12. REFERENSI

- [Dokumentasi Laravel](#)
 - [PEMECAHAN MASALAH.md](#)
 - [DEPLOYMENT.md](#)
 - [KEAMANAN.md](#)
-

Pemelihara: Tim Operasi TI

Kontak: ops@kopkarskf.com

Terakhir Ditinjau: 17 Januari 2026 Tinjauan

Berikutnya: 17 April 2026 (Triwulanan)