

## SISTEM ARSIP DIGITAL

Dinas Sosial

# MANUAL DOCKER

*Deploy & Manajemen di Docker sebagai Server Lokal/Komputer*

v1.0.0 | 2025

# TENTANG DOCKER DEPLOYMENT

---

Docker memungkinkan Anda menjalankan aplikasi Arsip Digital sebagai container yang terisolasi di komputer manapun, tanpa perlu menginstall Node.js atau dependencies lainnya secara manual. Ini adalah metode deployment yang direkomendasikan untuk server lokal kantor.

Keuntungan menggunakan Docker:

- Tidak perlu install Node.js secara manual
  - Mudah di-backup dan dipindahkan ke komputer lain
  - Data tersimpan di luar container (persistent)
  - Bisa dijalankan sebagai layanan yang otomatis start saat komputer dinyalakan
- 

## PRASYARAT

---

### Instalasi Docker

Docker Desktop harus terinstall di komputer/server yang akan digunakan:

- **Windows 10/11:** Download Docker Desktop dari <https://docker.com/products/docker-desktop>. Pastikan WSL2 diaktifkan.
- **Ubuntu/Linux:** Jalankan: `sudo apt update && sudo apt install docker.io docker-compose`
- **macOS:** Download Docker Desktop dari <https://docker.com/products/docker-desktop>

Verifikasi instalasi:

```
docker --version  
docker-compose --version
```

---

## STRUKTUR FILE DOCKER

---

### File Dockerfile

File Dockerfile sudah tersedia di root project. Berikut penjelasannya:

```
FROM node:20-alpine AS builder    # Tahap build  
WORKDIR /app  
COPY package*.json ./  
RUN npm install  
COPY . .  
RUN npm run build  
  
FROM node:20-alpine                # Tahap production  
WORKDIR /app  
ENV NODE_ENV=production  
COPY --from=builder /app/public ./public
```

```
COPY --from=builder /app/.next/standalone ./
RUN mkdir -p /app/data           # Folder untuk database
EXPOSE 3000
CMD ["node", "server.js"]
```

## File docker-compose.yml

File docker-compose.yml untuk manajemen yang mudah:

```
version: '3.9'
services:
  app:
    build: .
    container_name: arsip_app
    ports:
      - "3000:3000"
    volumes:
      - ./data:/app/data      # Data persisten di luar container
    env_file:
      - .env.docker
    restart: unless-stopped
```

## File .env.docker

Konfigurasi environment khusus untuk Docker:

```
DATABASE_URL=/data/arsip_dinsos.db
NODE_ENV=production
USE_HTTPS=false
```

---

# LANGKAH 1: PERSIAPAN

---

## 1.1 Pastikan File Konfigurasi Ada

Sebelum build Docker, pastikan next.config.ts memiliki output standalone:

```
const nextConfig = {
  output: 'standalone',
};
export default nextConfig;
```

## 1.2 Buat Folder Data

Folder data/ harus ada sebelum container dijalankan:

```
mkdir -p data
```

Jika ada database existing, salin ke folder ini:

```
cp /path/to/existing/arsip_dinsos.db ./data/arsip_dinsos.db
```

## LANGKAH 2: BUILD IMAGE DOCKER

Build Docker image dari source code:

```
docker-compose build
```

Atau menggunakan docker build langsung:

```
docker build -t arsip-digital:latest .
```

Proses build akan memakan waktu 5-15 menit pada build pertama karena mengunduh base image Node.js dan menginstall dependencies. Build selanjutnya lebih cepat karena cache.

*Jika ada perubahan pada kode sumber, Anda harus build ulang image: docker-compose build --no-cache*

## LANGKAH 3: MENJALANKAN CONTAINER

### 3.1 Jalankan dengan Docker Compose (Rekomendasi)

```
docker-compose up -d
```

Flag -d berarti detached mode (berjalan di background). Aplikasi akan tersedia di <http://localhost:3000>

### 3.2 Cek Status Container

```
docker-compose ps
```

Output yang diharapkan: STATUS = Up

### 3.3 Lihat Log Real-time

```
docker-compose logs -f
```

Tekan Ctrl+C untuk berhenti memantau log.

### 3.4 Akses dari Jaringan Lokal

Untuk mengizinkan komputer lain di jaringan mengakses, ubah port binding di docker-compose.yml:

```
ports:  
  - "0.0.0.0:3000:3000"
```

Komputer lain bisa akses via: [http://IP\\_KOMPUTER\\_SERVER:3000](http://IP_KOMPUTER_SERVER:3000)

Temukan IP komputer server dengan:

```
ipconfig    # Windows  
ip addr     # Linux
```

## MANAJEMEN CONTAINER

---

### Perintah-Perintah Penting

Perintah	Fungsi
docker-compose up -d	Start container di background
docker-compose down	Stop dan hapus container (data tetap aman)
docker-compose restart	Restart container
docker-compose logs -f	Monitor log real-time
docker-compose ps	Cek status container
docker exec -it arsip_app sh	Masuk ke dalam container (shell)
docker-compose pull	Update image dari registry

### Update Aplikasi

Ketika ada update source code:

1. Berhentikan container: docker-compose down
2. Backup database terlebih dahulu (salin folder data/)
3. Build ulang image: docker-compose build
4. Jalankan container baru: docker-compose up -d

---

## AUTO START SAAT BOOTING

---

### Windows - Menggunakan Task Scheduler

5. Buka Task Scheduler (cari di Start Menu)
6. Klik Create Basic Task
7. Nama: Arsip Digital Docker
8. Trigger: When the computer starts
9. Action: Start a program
10. Program: docker-compose
11. Arguments: -f C:\path\to\project\docker-compose.yml up -d

### Linux - Menggunakan Systemd

Buat file service:

```
sudo nano /etc/systemd/system/arsip-digital.service
```

Isi file:

```
[Unit]
Description=Arsip Digital Docker
After=docker.service
[Service]
WorkingDirectory=/path/to/project
ExecStart=/usr/bin/docker-compose up
ExecStop=/usr/bin/docker-compose down
Restart=always
[Install]
WantedBy=multi-user.target
```

Aktifkan service:

```
sudo systemctl enable arsip-digital
sudo systemctl start arsip-digital
```

---

## BACKUP & RESTORE DATABASE

---

### Backup Manual

Data tersimpan di folder data/ di host, bukan di dalam container. Untuk backup:

```
cp -r ./data ./data_backup_$(date +%Y%m%d)
```

### Backup via Aplikasi

Login ke aplikasi dan gunakan tombol Backup Database di Dashboard. File .db akan diunduh ke komputer Anda.

### Restore Database

12. Stop container: docker-compose down
13. Replace file database: cp backup.db ./data/arsip\_dinsos.db
14. Start container: docker-compose up -d

---

## TROUBLESHOOTING DOCKER

---

Masalah	Solusi
Port 3000 sudah terpakai	Ubah port di docker-compose.yml: '3001:3000'

Masalah	Solusi
Container tidak bisa start	Cek log: docker-compose logs. Pastikan folder data/ ada
Database error	Cek permission folder data/: chmod 777 ./data
Image build gagal	Cek next.config.ts sudah ada output:'standalone'
Perubahan kode tidak terlihat	Build ulang: docker-compose build --no-cache && docker-compose up -d