

Docker

Docker

Projekt używa **Docker Desktop** z wieloetapowym `Dockerfile` i `docker-compose.yml` opartym na profilach.

Profile

Profil	Kontener	Port	Opis
dev	combo_dev	5173	Vite hot-reload, pliki montowane z hosta
prod	combo_prod	4173	nginx serwujący zbudowany <code>dist/</code>

Komendy

Uruchomienie — tryb dev

```
docker compose --profile dev up
```

Aplikacja dostępna pod **http://localhost:5173**

Każda zmiana w `src/` lub `public/data/` jest widoczna w przeglądarce natychmiast bez restartu kontenera.

Uruchomienie — tryb prod

```
docker compose --profile prod up --build
```

Aplikacja dostępna pod **http://localhost:4173**

Zatrzymanie

```
docker compose --profile dev down  
# lub  
docker compose --profile prod down
```

Rebuild po zmianie zależności (package.json)

```
docker compose --profile dev build --no-cache  
docker compose --profile dev up
```

Podgląd logów (live)

```
docker logs combo_dev -f
```

Shell wewnątrz kontenera

```
docker exec -it combo_dev sh
```

Sieć

Projekt używa izolowanej sieci `combo_net` (bridge, `172.23.0.0/16`).

Nie koliduje z istniejącymi kontenerami WordPress:

Kontener	Port	Sieć
<code>spzw_wp</code>	8080	<code>spzw_frontend</code>
<code>spzw_pma</code>	8081	<code>spzw_frontend</code>
<code>spzw_db</code>	3306	<code>spzw_backend</code>
<code>combo_dev</code>	5173	<code>combo_net</code>
<code>combo_prod</code>	4173	<code>combo_net</code>

Struktura plików Docker

```
combo-raport-app/
├─ Dockerfile          # 5 etapów: base → deps → dev / build → prod
├─ docker-compose.yml  # profile dev i prod
├─ .dockerignore       # wyklucza node_modules, dist, .git, .claude
└─ docker/
    └─ nginx.conf      # SPA routing, cache assetów, nagłówki bezpieczeństwa
```

Dockerfile — etapy

```
base (node:22-alpine + pnpm via corepack)
├─ deps (pnpm install --frozen-lockfile, cache mount)
│   ├── dev    → EXPOSE 5173, CMD vite --host 0.0.0.0
│   ├── build  → RUN pnpm build
│   └─ prod    → nginx:1.27-alpine, EXPOSE 4173
```

Etap `prod` kopiuje tylko `dist/` — nie zawiera `node_modules` ani kodu źródłowego.

nginx.conf — kluczowe zachowania

- **SPA routing** — wszystkie ścieżki przekierowywane do `index.html` (`try_files $uri /index.html`)
- **Cache assetów** — JS/CSS/fonty/obrazy: `Cache-Control: public, immutable, 1 rok`
- **Brak cache dla index.html** — `no-cache, no-store, must-revalidate`
- **Kompresja gzip** — włączona dla JS, CSS, JSON, SVG
- **Nagłówki bezpieczeństwa** — `X-Frame-Options`, `X-Content-Type-Options`, `Referrer-Policy`

Wskazówki

Zmiana portu dev — edytuj `docker-compose.yml` (`"5173:5173"`) i dodaj `server: { port: XXXX }` w `vite.config.ts`, następnie zrób rebuild.

Logi w Docker Desktop — zakładka *Containers* → kliknij `combo_dev` → zakładka *Logs*.

Uruchomienie bez Dockera (lokalnie) — patrz [Struktura projektu](#) → [Skrypty pnpm](#).

