PROGRAM SZKOLENIA NARZĘDZIA I DEVOPS

Szkolenie Docker

Uruchomisz aplikację z bazą danych jedną komendą i zbudujesz lekki, bezpieczny obraz Dockera.

CZAS TRWANIA	POZIOM	FORMA
2 dni	**od podstaw**	**warsztat dla zespołu, zdalnie lub stacjonarnie**

„U mnie działa" przestaje być problemem, gdy aplikacja i jej zależności startują w kontenerach. Na szkoleniu uczysz się Dockera od strony programisty: od pierwszego `Dockerfile` po lokalne środowisko z bazą danych uruchamiane jedną komendą.

Dowiesz się też, jak budować obrazy szybko i tak, żeby były małe i bezpieczne, bo te same obrazy trafiają potem do CI i na produkcję. Kubernetes zostaje poza zakresem - te dwa dni przygotowują Cię do niego.

PROGRAM

2 dni

- Po co kontenery
  - Kontener a maszyna wirtualna
  - Obraz a kontener
  - Rejestry obrazów: Docker Hub i rejestry prywatne
  - Licencje w firmie: Docker Desktop płatny powyżej 250 pracowników lub 10 mln USD przychodu, OrbStack przy każdym użyciu komercyjnym, Podman i Colima bezpłatne
- Praca z kontenerami
  - `docker run`, `ps`, `logs`, `exec`, `stop`
  - Mapowanie portów i zmienne środowiskowe
  - Sprzątanie: nieużywane obrazy, kontenery i wolumeny
- Dockerfile
  - Instrukcje `FROM`, `COPY`, `RUN`, `CMD`, `ENTRYPOINT`
  - Warstwy i cache: kolejność instrukcji ma znaczenie
  - Plik `.dockerignore`
  - BuildKit i `docker buildx`: obrazy dla arm64 i amd64
- Multi-stage build
  - Osobny etap budowania i uruchamiania
  - Obrazy bazowe: `alpine`, `slim`, `distroless` i darmowe Docker Hardened Images

- Rozmiar i bezpieczeństwo obrazu
  - Uruchamianie procesu jako użytkownik bez uprawnień roota
  - Skanowanie obrazu pod kątem podatności: Docker Scout i Trivy
  - Przypinanie wersji obrazów bazowych i narzędzi: lekcja z ataku na Trivy w 2026 roku
- Dane i sieć
  - Wolumeny i bind mounty
  - Sieci między kontenerami i rozwiązywanie nazw
- Docker Compose
  - Aplikacja i baza danych w jednym pliku `compose.yaml`
  - Healthcheck i `depends_on` z warunkiem `service_healthy`
  - `docker compose watch`: synchronizacja plików i przebudowa bez ręcznego restartu
- Konfiguracja i sekrety
  - Pliki `.env` i zmienne w Compose
  - Sekrety przy budowaniu: `RUN --mount=type=secret` zamiast `ARG`
  - Sekrety w czasie uruchamiania: `secrets:` w Compose zamiast zmiennych środowiskowych
- Debugowanie kontenerów
  - Wejście do działającego kontenera
  - Czytanie logów i stanu kontenera
  - Kontener, który od razu się zamyka
  - `lazydocker`: kontenery, logi i statystyki w jednym oknie terminala
- Obraz gotowy do CI
  - Tagowanie: wersja i SHA commita zamiast samego `latest`
  - Publikacja w rejestrze: GitHub Container Registry i Docker Hub

Cache warstw między przebiegami i gotowy workflow, który buduje i publikuje obraz, pokazuję na szkoleniu CI/CD z GitHub Actions.

DLA KOGO?

- Dla programistów, którzy chcą uruchamiać projekt bez ręcznej instalacji zależności.
- Dla zespołów, w których każdy ma inną wersję bazy danych albo środowiska uruchomieniowego.
- Dla osób, które zaczynają przygodę z CI i wdrożeniami i chcą przygotować do nich obraz.

CO MUSISZ UMIEĆ?

- Pracować w terminalu na poziomie podstawowym. Dockera nie musisz znać.
- Uruchomić dowolną aplikację webową lokalnie.
- Mieć komputer z Dockerem (Docker Desktop, OrbStack albo Colima z Docker CLI) i uprawnieniami do instalacji programów. Na Podmanie przejdziesz podstawy, ale buildx, Docker Scout i `compose watch` pokazuję na Dockerze.
- Posługiwać się Gitem i GitHubem na poziomie podstawowym: commit, push, pull request.

CZEGO SIĘ NAUCZYSZ?

- Wybierzesz środowisko do kontenerów zgodne z licencją w Twojej firmie.
- Napiszesz `Dockerfile` z multi-stage build i dobrze wykorzystanym cache.
- Zbudujesz obraz, który działa i na Macu z Apple Silicon, i na serwerze amd64.
- Uruchomisz aplikację z bazą danych przez Docker Compose i zobaczysz zmiany w kodzie bez ręcznego restartu.
- Przekażesz sekret do budowania obrazu tak, żeby nie został w warstwach.
- Znajdziesz przyczynę, dla której kontener nie startuje.
- Zmniejszysz rozmiar obrazu, ograniczysz jego uprawnienia i sprawdzisz go pod kątem podatności.
- Przygotujesz obraz gotowy do budowania i publikacji w pipeline CI.

OPINIE UCZESTNIKÓW

> *„Gruntownie, krok po kroku stworzyliśmy aplikację sklepu internetowego, po drodze pisząc testy jednostkowe, e2e, web komponenty, autorski backend oparty na Node.js + Express.js, jak również wykorzystaliśmy narzędzia CI."*
>
> Łukasz, JavaScript Developer (WarsawJS MasterClass)

> *„Piotr przekazywał wiedzę na temat tworzenia odpornych na błędy aplikacji internetowych. Podczas szkolenia tworzyliśmy testy jednostkowe, testy end-to-end aplikacji po stronie klienta oraz testy backendu napisanego w Node.js."*
>
> Dominik, Frontend Developer (WarsawJS MasterClass)

TRENER

Piotr Kowalski

W branży IT od 2008 roku. Od 2014 jest mentorem dla wielu developerów. Jako trener spędził na sali szkoleniowej ponad 1000 godzin, a na LinkedIn ma 52 rekomendacje. Każde jego szkolenie to dawka solidnej wiedzy oraz humoru.

Więcej: piecioshka.pl/piotr-kowalski

ZORGANIZUJMY TO SZKOLENIE W TWOIM ZESPOLE

Program dopasuję do Waszego projektu i poziomu grupy. Napisz, z czym pracujecie, a odpowiem z propozycją terminu i zakresu.

KONTAKT **piecioshka.pl/kontakt**
E-MAIL **piecioshka+blog [małpka] gmail.com**
SZKOLENIE **piecioshka.pl/szkolenia/docker/**