



Kompleksowe szkolenie - Konteneryzacja Docker i Kubernetes: od zera do bohatera

Numer usługi 2026/02/07/202247/3315236

5 535,00 PLN brutto

4 500,00 PLN netto

158,14 PLN brutto/h

128,57 PLN netto/h

332,00 PLN cena rynkowa ⓘ

JSYSTEMS SPÓŁKA
Z OGRANICZONĄ
ODPOWIEDZIALNOŚ
CIĄ

Brak ocen dla tego dostawcy

- 📄 Usługa szkoleniowa
- 📺 zdalna w czasie rzeczywistym
- 🕒 35:00 h
- 📅 20.07.2026 do 24.07.2026

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Administracja IT i systemy komputerowe
Grupa docelowa usługi	Skierowane do programistów i devOpsów przeznaczone dla osób nieznających tematyki kontenerów i zarządzania nimi.
Minimalna liczba uczestników	6
Maksymalna liczba uczestników	15
Data zakończenia rekrutacji	16-07-2026
Forma prowadzenia usługi	zdalna w czasie rzeczywistym
Liczba godzin usługi	35
Podstawa uzyskania wpisu do BUR	Certyfikat systemu zarządzania jakością wg. ISO 9001:2015 (PN-EN ISO 9001:2015) - w zakresie usług szkoleniowych

Cel

Cel edukacyjny

Nabycie przez uczestników kompleksowych umiejętności praktycznych w zakresie konteneryzacji aplikacji przy użyciu Docker i orkiestracji kontenerów w Kubernetes, umożliwiających samodzielne wdrażanie, skalowanie i zarządzanie aplikacjami kontenerowymi w środowiskach produkcyjnych.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
<p>Tworzy obrazy Docker i zarządza kontenerami.</p> <p>Konfiguruje wielokontenerowe aplikacje przy użyciu Docker Compose.</p>	<p>Uczestnik pisze Dockerfile dla wskazanej aplikacji, buduje obraz, uruchamia kontener i weryfikuje dostępność aplikacji w przeglądarce.</p> <p>Uczestnik tworzy plik docker-compose.yml definiujący aplikację wielousługową (np. aplikacja + baza danych) i uruchamia środowisko jednym poleceniem z poprawnymi zależnościami.</p>	<p>Test teoretyczny z wynikiem generowanym automatycznie</p> <p>Test teoretyczny z wynikiem generowanym automatycznie</p>
<p>Wdraża aplikacje w klastrze Kubernetes (Deployments, Services, ConfigMaps).</p>	<p>Uczestnik tworzy manifesty Kubernetes (Deployment + Service) wdrażające aplikację kontenerową w klastrze i weryfikuje jej dostępność przez ClusterIP/NodePort.</p>	<p>Test teoretyczny z wynikiem generowanym automatycznie</p>
<p>Konfiguruje skalowanie i aktualizacje rolling update w Kubernetes.</p>	<p>Uczestnik skaluje deployment do wskazanej liczby replik, przeprowadza rolling update do nowej wersji obrazu i weryfikuje brak przestoju podczas aktualizacji.</p>	<p>Test teoretyczny z wynikiem generowanym automatycznie</p>

Kwalifikacje

Kompetencje

Usługa prowadzi do nabycia kompetencji.

Warunki uznania kompetencji

Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji lub wyraźnie z nim powiązane inne dokumenty związane ze wsparciem zawierają opis efektów uczenia się?

TAK

Pytanie 2. Czy dokument lub wyraźnie z nim powiązane inne dokumenty związane ze wsparciem potwierdzają, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji i zgodnie z zaplanowanymi metodami walidacji?

TAK

Pytanie 3. Czy dokument lub wyraźnie z nim powiązane inne dokumenty związane ze wsparciem potwierdzają zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji?

TAK

Program

1. Wprowadzenie

2. Co to jest mikroserwis

3. REST API

- o Wprowadzenie do REST API

- o czy to jedyna opcja komunikacji między kontenerami?

4. Dobre praktyki - 12 factor app

- o Przegląd dobrych praktyk przy tworzeniu aplikacji

- o Przykłady złych aplikacji

5. Docker - co to jest, gdzie pływa i gdzie dokuje?

- o Instalacja i konfiguracja

- o Docker images - czym są obrazy, tworzenie i obsługa

- o UnionFs

- o Uruchomienie Hello World!

- o Kontenery - tworzenie, współdzielenie, uruchamianie i podział warstw

6. Docker - tworzenie własnych obrazów

- o Budujemy własne obrazy i poznajemy Dockerfile

- o ENTRYPOINT vs CMD - jaka jest różnica i po co się to stosuje

- o .dockerignore - co jeżeli nie chcemy mieć niektórych rzeczy

- o Multistage build - staramy się trzymać małe obrazy

- o Optymalizacja warstw w Docker - jak budować dobre obrazy?

7. Registry - czyli repozytorium obrazów

- o Uruchamiamy lokalne registry

- o Zabezpieczamy dostęp do niego

- o Docker Hub - ściąganie i wysyłanie obrazów

- o Chmura czy lokalne registry (jakie są rozwiązania i które będzie dla mnie najlepsze)

8. Docker - deep dive

- o Budujemy obraz aplikacji z wykorzystaniem Maven i korzystamy z własnego rejestru

- o Docker kill vs stop vs pause

- o Alpine Linux

- o Uczymy się uruchamiać aplikacje na przykładzie aplikacji SpringBoot

- o Dopakowujemy kontenery -> VOLUME

- o Izolacja kontenerów

- o Nasłuch kontenerów i dostęp - EXPOSE, ENV, WORKDIR, USER

- o Kontenery a powłoka shellowa (Bash vs sh vs zsh)

o Budujemy obraz Nginx i uruchamiamy ponownie naszą apkę SpringBoot

o Debugowanie kontenerów

o Docker in Docker - co to i czy warto?

9. Security i inne

o Bezpieczeństwo kontenerów i skanowanie pod kątem bezpieczeństwa.

o Jak zrobić code review – mikroserwisów

10. Docker Compose

o Wprowadzenie i instalacja

o Konfiguracja i format YAML

o Izolacja kontenerów

o Przeładowywanie i modyfikacja kontenerów - wykonujemy update

o Wstrzykiwanie zmiennych środowiskowych

o Łatwe skalowanie kontenerów z wykorzystaniem traefika

o Uruchamiamy apkę SpringBoot z wykorzystaniem JPA

o Dodajemy bazę danych

o Produkcyjne Docker Compose - najlepsze praktyki

o Logowanie w docker na przykładzie sysloga i Elastic Stack

o Skalujemy nasze aplikacje - jak zeskalować nasze aplikacje

11. Portainer - graficzny interfejs użytkownika - co to jest i czy warto

12. Kubernetes

o Podstawy architektury

1. API

2. Opis obiektów

o Omówienie kubectl i podobieństwa z docker

1. Wejście w POD i kontener

2. Zarządzanie kontekstem

o Pod:

1. Lifecycle kontenera - co się dzieje jak POD nie działa?

2. Zarządzanie zasobami - jak ograniczyć zasoby naszego kontenera

3. Healthcheck - co to jest i jak go stosować

4. Wiele kontenerów w Pod - czy można?

5. Przeglądanie logów i eventów - co jak idzie coś nie tak..

6. Selectory i Adnotacje - czy warto to stosować?

o Service discovery w docker:

1. Service discovery w Docker

2. DNS

o Repliki, skalowanie a deployment wersji

1. Recreate i rolling update - jak wdrażać nasze aplikacje na produkcję
2. Manualne i automatyczne skalowanie aplikacji - co zrobić jak nasza aplikacja potrzebuje więcej zasobów czy może się sama zeskalować

o Eksponowanie usług

1. NodePort
2. ClusterIP
3. LoadBalancer
4. Ingress
 1. na przykładzie NGINX i Traefik
 2. używanie certyfikatów w ingress
 3. automatyczne generowanie certyfikatów na przykładzie Let's encrypt

o Konfiguracja i współdzielenie informacji

1. Sekrety
2. ConfigMaps
3. Dobre praktyki na produkcji - wykorzystanie Hashicorp Vault

o Job i CronJob - zadania wsadowe i cykliczne

o Persistent storage - montowanie dysków

o DemonSets

o StatefulSets - czyli bazy danych w docker - stosować czy nie? Jak to robić?

13. Monitorowanie aplikacji

o Jakie są sposoby monitorowania Kubernetesa?

o Prometheus czy jest to najlepsze narzędzie?

o Dashboard kubernetesowy

14. Centralne systemy logowania

o Jakie są sposoby logowania

o Który jest najlepszy?

15. Jak postawić klaster od zera - jakie są możliwości?

16. Kilka przydatnych rzeczy:

o Helm

o Kustomize

o KIND

17. Security w Kubernetes

18. CI i Code repository

o Jak robić Code Review mikroserwisów

o Ułożenie Continuous Integration pipeline

Harmonogram

Liczba pozycji harmonogramu: 15

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 15 Wprowadzenie	Maciej Krajewski	20-07-2026	09:00	12:30	03:30
2 z 15 Przerwa Obiadowa	Maciej Krajewski	20-07-2026	12:30	13:00	00:30
3 z 15 Co to jest mikroserwis	Maciej Krajewski	20-07-2026	13:00	16:00	03:00
4 z 15 REST API/ Dobre praktyki - 12 factor app	Maciej Krajewski	21-07-2026	09:00	12:30	03:30
5 z 15 Przerwa Obiadowa	Maciej Krajewski	21-07-2026	12:30	13:00	00:30
6 z 15 Docker - co to jest, gdzie plywa i gdzie dokuje?/Docker - tworzenie własnych obrazów	Maciej Krajewski	21-07-2026	13:00	16:00	03:00
7 z 15 Registry - czyli repozytorium obrazów/Docker - deep dive	Maciej Krajewski	22-07-2026	09:00	12:30	03:30
8 z 15 Przerwa Obiadowa	Maciej Krajewski	22-07-2026	12:30	13:00	00:30
9 z 15 Security i inne/Docker Compose	Maciej Krajewski	22-07-2026	13:00	16:00	03:00
10 z 15 Portainer - graficzny interfejs użytkownika - co to jest i czy warto/Kubernet e	Maciej Krajewski	23-07-2026	09:00	12:30	03:30
11 z 15 Przerwa Obiadowa	Maciej Krajewski	23-07-2026	12:30	13:00	00:30

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
12 z 15 Monitorowanie aplikacji/Centralne systemy logowania	Maciej Krajewski	23-07-2026	13:00	16:00	03:00
13 z 15 Jak postawić klaster od zera - jakie są możliwości?/Kilka przydatnych rzeczy:	Maciej Krajewski	24-07-2026	09:00	12:30	03:30
14 z 15 Przerwa Obiadowa	Maciej Krajewski	24-07-2026	12:30	13:00	00:30
15 z 15 Security w Kubernetes/CI i Code repository	Maciej Krajewski	24-07-2026	13:00	15:30	02:30

Cennik

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	5 535,00 PLN
Koszt przypadający na 1 uczestnika netto	4 500,00 PLN
Koszt osobogodziny brutto	158,14 PLN
Koszt osobogodziny netto	128,57 PLN

Prowadzący

Liczba prowadzących: 1



1 z 1

Maciej Krajewski

Trener posiada wieloletnie doświadczenie w obszarze architektury i administracji środowisk kontenerowych oraz infrastruktury chmurowej, przy czym kluczowe kwalifikacje w zakresie Docker, Kubernetes, DevOps i automatyzacji procesów CI/CD zostały zdobyte i są czynnie wykorzystywane w okresie ostatnich 5 lat (od 2021 roku do chwili obecnej). Potwierdzają to zrealizowane projekty

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Informacje o materiałach dla uczestników usługi - Uczestnicy otrzymają komplet materiałów PDF. Każdy uczestnik otrzymuje kod dostępu i

dane logowania do platformy ZOOM na 7 dni przed datą rozpoczęcia szkolenia. Dane

przesyłane są na adres e-mail podany podczas rejestracji.

Warunki uczestnictwa

Znajomość podstaw programowania oraz podstawowa orientacja w pracy z systemem Linux i terminalem, umiejętność korzystania z komputera

Informacje dodatkowe

Warunkiem ukończenia szkolenia i otrzymania zaświadczenia jest uzyskanie minimalnej

frekwencji na poziomie 80% całkowitego czasu trwania usługi. Obecność uczestnika będzie

potwierdzana na podstawie codziennych list obecności lub logów z platformy online.

Warunki techniczne

Uczestnik musi dysponować sprzętem i łączem o parametrach:

- Procesor: min. 4-rdzeniowy (np. Intel i5/i7 lub odpowiednik AMD/M1/M2)
- Pamięć RAM: min. 16 GB
- Dysk: min. 20 GB wolnej przestrzeni
- System operacyjny: Windows 10/11 Pro, Linux lub macOS
- Multimedia: Sprawna kamera internetowa oraz mikrofon (wymagane do komunikacji i weryfikacji obecności)
- Łącze internetowe: Stabilne połączenie o minimalnej prędkości 10 Mbps (download) / 5 Mbps (upload)
- Oprogramowanie: Uprawnienia administratora pozwalające na instalację narzędzi

Kontakt



Biuro Obsługi Klienta

E-mail biuro@jssystemy.pl

Telefon (+48) 534 506 503