



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

Numer usługi 2026/02/03/202247/3305311

5 535,00 PLN brutto

4 500,00 PLN netto

158,14 PLN brutto/h

128,57 PLN netto/h

JSYSTEMS SPÓŁKA  
Z OGRANICZONĄ  
ODPOWIEDZIALNOŚ  
CIĄ

Brak ocen dla tego dostawcy

📍 zdalna w czasie rzeczywistym

🛠 Usługa szkoleniowa

🕒 35 h

📅 25.05.2026 do 29.05.2026

## Informacje podstawowe

<b>Kategoria</b>	Informatyka i telekomunikacja / Administracja IT i systemy komputerowe
<b>Grupa docelowa usługi</b>	Skierowane do programistów i devOpsów przeznaczone dla osób nieznających tematyki kontenerów i zarządzania nimi.
<b>Minimalna liczba uczestników</b>	6
<b>Maksymalna liczba uczestników</b>	15
<b>Data zakończenia rekrutacji</b>	21-05-2026
<b>Forma prowadzenia usługi</b>	zdalna w czasie rzeczywistym
<b>Liczba godzin usługi</b>	35
<b>Podstawa uzyskania wpisu do BUR</b>	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 zawiera opis efektów uczenia się?

TAK

Pytanie 2. Czy dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji?

TAK

Pytanie 3. Czy dokument potwierdza 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. Lifecyle 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 przedmiotów/zajęć: 16

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
<b>1 z 16</b> Wprowadzenie/C o to jest mikroserwis	Maciej Krajewski	25-05-2026	09:00	12:30	03:30
<b>2 z 16</b> Przerwa Obiadowa	Maciej Krajewski	25-05-2026	12:30	13:00	00:30
<b>3 z 16</b> REST API/Dobre praktyki - 12 factor app	Maciej Krajewski	25-05-2026	13:00	16:00	03:00
<b>4 z 16</b> Docker - co to jest, gdzie pływa i gdzie dokuje?/Docker - tworzenie własnych obrazów	Maciej Krajewski	26-05-2026	09:00	12:30	03:30
<b>5 z 16</b> Przerwa Obiadowa	Maciej Krajewski	26-05-2026	12:30	13:00	00:30
<b>6 z 16</b> Registry - czyli repozytorium obrazów/Docker - deep dive	Maciej Krajewski	26-05-2026	13:00	16:00	03:00
<b>7 z 16</b> Security i inne/Docker Compose	Maciej Krajewski	27-05-2026	09:00	12:30	03:30
<b>8 z 16</b> Przerwa Obiadowa	Maciej Krajewski	27-05-2026	12:30	13:00	00:30
<b>9 z 16</b> Portainer - graficzny interfejs użytkownika - co to jest i czy warto/Kubernetes	Maciej Krajewski	27-05-2026	13:00	16:00	03:00

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
<b>10 z 16</b> Monitorowanie aplikacji/Centralne systemy logowania	Maciej Krajewski	28-05-2026	09:00	12:30	03:30
<b>11 z 16</b> Przerwa Obiadowa	Maciej Krajewski	28-05-2026	12:30	13:00	00:30
<b>12 z 16</b> Jak postawić klaster od zera - jakie są możliwości?	Maciej Krajewski	28-05-2026	13:00	16:00	03:00
<b>13 z 16</b> Security w Kubernetes/CI i Code repository	Maciej Krajewski	29-05-2026	09:00	12:30	03:30
<b>14 z 16</b> Przerwa Obiadowa	Maciej Krajewski	29-05-2026	12:30	13:00	00:30
<b>15 z 16</b> Zajęcia szkoleniowe – podsumowanie	Maciej Krajewski	29-05-2026	13:00	15:30	02:30
<b>16 z 16</b> Walidacja	Maciej Krajewski	29-05-2026	15:30	16:00	00:30

## 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 wdrożeniowe dla klientów komercyjnych oraz szkolenia z konteneryzacji i DevOps w latach 2022–2026.

# 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@jssystem.com](mailto:biuro@jssystem.com)

**Telefon** (+48) 534 506 503