



Kubernetes: orkiestracja kontenerów - szkolenie kompleksowe

Numer usługi 2026/02/13/202247/3333620

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

📅 07.12.2026 do 11.12.2026

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Administracja IT i systemy komputerowe
Grupa docelowa usługi	Skierowane do deweloperów i administratorów, a także wszystkich zainteresowanych tworzeniem, uruchamianiem i zarządzaniem aplikacjami kontenerowymi.
Minimalna liczba uczestników	6
Maksymalna liczba uczestników	15
Data zakończenia rekrutacji	03-12-2026
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

Po ukończeniu tego szkolenia uczestnik będzie potrafił: • tworzyć i optymalizować kontenery – efektywne zarządzanie konfiguracjami i bezpieczeństwem,

• administrować Kubernetes – kontrolować obiekty, zarządzać zasobami i wdrażać aplikacje,

• skalować i monitorować aplikacje – wdrażać strategie aktualizacji i automatycznego skalowania,

- zabezpieczać klaster – stosować polityki bezpieczeństwa i kontrolować dostęp,
- efektywnie wdrażać aplikacje – używać Helm, Kustomize oraz nowoczesne praktyki

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
<p>Po ukończeniu tego szkolenia uczestnik będzie potrafił:</p> <ul style="list-style-type: none"> • tworzyć i optymalizować kontenery – efektywne zarządzanie konfiguracjami i bezpieczeństwem, • administrować Kubernetes – kontrolować obiekty, zarządzać zasobami i wdrażać aplikacje, • skalować i monitorować aplikacje – wdrażać strategie aktualizacji i automatycznego skalowania, • zabezpieczać klaster – stosować polityki bezpieczeństwa i kontrolować dostęp, • efektywnie wdrażać aplikacje – używać Helm, Kustomize oraz nowoczesne praktyki DevOps. 	<p>Każde omawiane zagadnienie jest utrwalane rzędem ćwiczeń o wzrastającym poziomie trudności. Podczas szkolenia uczestnicy wykonują ponad 50 warsztatów. W szkoleniu duży nacisk jest kładziony na dobre zrozumienie zasad działania wykorzystywanych technik, a nie ich odtwórcze stosowanie.</p>	<p>Obserwacja w warunkach rzeczywistych</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

o Architektura aplikacji kontenerowych - Docker vs Portainer vs ContainerD - czy to ma znaczenie?

- o Optymalizacja Dockerfile, jak to robić dobrze
- o Bezpieczeństwo kontenerów jak o nie zadbać i jak je skanować?
- o Hardening obrazów - czyli jak dodatkowo zabezpieczyć obraz
- o Jak dobrze zarządzać konfiguracjami w dockerze na przykładzie docker compose
- o Administracja kontenerami - na co zwrócić uwagę, skonfigurowanie logowania, metryki, tracing, itp.
- o Narzędzia wspierające codzienną pracę z dockerem.

2. Kubernetes

- o Podstawy architektury

- API

- Opis obiektów

- o Zarządzanie obiektami Kubernetes (imperatywne oraz deklaratywne)

- o Manifesty obiektów - struktura oraz format (YAML)

- o Omówienie kubectl i podobieństwa z docker

- Wejście w POD i kontener

- Zarządzanie kontekstem

- o Pod

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

- Uruchamianie kontenerów/zadań przed startem głównej aplikacji

- Zarządzanie zasobami - jak ograniczyć zasoby naszego kontenera

- Healthcheck - co to jest i jak go stosować (Liveness, Readiness oraz Startup probes)

- Wiele kontenerów w Pod - czy można?

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

- Label, Selector i Adnotacje - jakie label stosować, aby się nie pogubić

- o Service discovery w Kubernetes

- Service discovery w Kubernetes

- DNS

- o Repliki, skalowanie a deployment wersji

- Recreate i rolling update - jak wdrażać nasze aplikacje na produkcję

- Omówienie polityk aktualizacji aplikacji (recreate, ramped, blue/green, canary, a/b testing, shadow)

- Canary Release - jak to dobrze zrobić na Kubernetesie (oraz jak zrobić to źle)

- Manualne i automatyczne skalowanie aplikacji - co zrobić jak nasza aplikacja potrzebuje więcej zasobów czy może się sama zeskalować - HPA

- Startowanie aplikacji na określonych maszynach oraz konfiguracja zależności pomiędzy uruchomionymi kontenerami (nodeSelector, affinity/antiAffinity, taints/tolerations)

- Zarządzanie zasobami klastra: minimalnymi oraz maksymalnymi limitami przypisanymi do kontenerów

- Priorytety aplikacji w klastrze oraz wyłączenie kontenerów z niskim priorytetem

- Utrzymanie maszyn klastra wraz z przygotowaniem okien obsługi

o Namespaces - wirtualne przestrzenie

o Eksponowanie usług

☒ NodePort

☒ ClusterIP

☒ LoadBalancer

☒ Ingress

☒ NGINX ingress controller i alternatywy

☒ Zarządzanie routowaniem aplikacji

☒ Używanie certyfikatów w ingress

☒ Automatyczne generowanie certyfikatów na przykładzie Let's encrypt i Cert Managera

o Konfiguracja i współdzielenie informacji

☒ Sekrety

☒ ConfigMaps

☒ Dobre praktyki na produkcji - wykorzystanie Hashicorp Vault

o Job i CronJob - zadania wsadowe i cykliczne

o Storage

☒ Lokalne montowanie danych

☒ EmptyDir

☒ Zarządzanie zmiennymi środowiskowymi i plikami konfiguracyjnymi aplikacji z użyciem ConfigMaps

☒ Przechowywanie danych wrażliwych takich jak: hasła, klucze czy tokeny za pomocą Secrets (generic, docker-registry, tls)

☒ Utrwalanie danych z użyciem wolumenów różnego typu

☒ Dynamiczne oraz statyczne zarządzanie wolumenami w klastrze

o DemonSets

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

o QoS

3. Uwierzytelnianie oraz autoryzacja

o Organizacja informacji o klastrach i użytkownikach w pliku kubeconfig

o Przedstawienie typów użytkowników w Kubernetes: używanych przez administratorów (użytkownicy) oraz aplikacje (konta serwisowe)

o Omówienie strategii uwierzytelniania w klastrze, od użytkownika z hasłem, przez certyfikaty x509, po tokeny OpenID

o Kontrola dostępu oparta na rolach jako sposób na przypisywanie uprawnień do użytkowników

o Walidacja lub modyfikacja żądań za pomocą Admission Controllers

4. Sieci

o CNI jako interfejs służący do konfiguracji kart sieciowych kontenerów

o Który CNI wybrać?

o Blokada komunikacji sieciowej w klastrze za pomocą Network Policies

o Mesh - co to jest

5. Monitorowanie aplikacji

- o Jakie są sposoby monitorowania Kubernetesa?
- o Prometheus czy jest to najlepsze narzędzie?
- o Dashboard kubernetesowy i alternatywy

6. Centralne systemy logowania

- o Omówienie różnych architektur zbierania logów w klastrze Kubernetes wraz z przeglądem najpopularniejszych narzędzi
- o Centralny monitoring zasobów w klastrze na przykładzie: Prometheus, AlertManager, Grafana

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

8. Kilka przydatnych rzeczy

- o Helm
- o Kustomize

9. Security w Kubernetes

- o Testy penetracyjne klastra Kubernetes
- o Ograniczenie uprawnień oraz kontrola dostępu aplikacji do komponentów systemu operacyjnego z użyciem SecurityContext
- o Wymuszanie globalnych standardów specyfikacji aplikacji wykorzystując Pod Security Policy

10. Dystrybucje oraz użyteczne narzędzia

- o Kubernetes jako usługa na przykładzie najpopularniejszych dostawców chmur publicznych (GKE, EKS, AKS)
- o Zautomatyzowana instalacja klastra przy użyciu Kubernetes Operations (KOPS)
- o Instalacja klastra Kubernetes "on premise" (Kubespray)
- o kubectl, kubens, telepresence, lens i inne narzędzia użyteczne w codziennej pracy z klastrem Kubernetes

Harmonogram

Liczba pozycji harmonogramu: 0

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
Brak wyników.					

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

Pasjonat kultury DEVOPS, zaprawiony w bojach programista JAVA i DEVOPS od 15 lat (jeszcze zanim to było modne). Ma wieloletnie doświadczenie w zarządzaniu wszelorakimi zespołami developerskimi. Zjadł zęby na realizacji projektów informatycznych w metodologiach klasycznych i zwinnych. Od systemów telekomunikacyjnych opartych na VoIP, rozwiązań video dla firm ubezpieczeniowych, banków, kilka fintechowych blockchainów, po centralny system do przetwarzania faktur medycznych dla rządu w Ghanie. Uwielbia proste rozwiązania i na codzień jest leniwy, dlatego jeżeli może automatyzuje co może - stąd wielka miłość do konteneryzacji i k8s'a, technologii CI/CD oraz CM! Maciej posiada wieloletnie doświadczenie trenerskie, Master of Ceremonies na scenie TECH konferencji Infoshare, na codzień zarządza holendersko-polską technologiczną Yameo

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Uczestnicy otrzymają komplet materiałów PDF

Warunki uczestnictwa

Praktyczna znajomość technologii kontenerowej np. Docker Wiedza z zakresu programowania, sieci komputerowych oraz systemu Linux, umiejętność korzystania z komputera

Informacje dodatkowe

Szkolenie kończy się wydaniem certyfikatu imiennego

Warunki techniczne

Dostęp do internetu, laptop/komputer, wideoaplikacja ZOOM lub dostęp do wersji w przeglądarce, możliwość podłączenia się do hostów przygotowanych na pulpicie zdalnym.

Kontakt



Biuro Obsługi Klienta

E-mail biuro@jssystem.com

Telefon (+48) 534 506 503