



Kompleksowe wprowadzenie do DevOps - szkolenie w formie warsztatowej

Numer usługi 2026/02/04/202247/3306769

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
- 📅 08.06.2026 do 12.06.2026

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Administracja IT i systemy komputerowe
Grupa docelowa usługi	Skierowane do początkujących DevOpsów, administratorów systemów, programistów oraz osób zainteresowanych automatyzacją, konteneryzacją i wdrażaniem aplikacji w nowoczesnych środowiskach.
Minimalna liczba uczestników	6
Maksymalna liczba uczestników	15
Data zakończenia rekrutacji	04-06-2026
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 wdrażania środowisk DevOps, zarządzania wersjami kodu oraz automatyzacji zadań administracyjnych, umożliwiających samodzielną pracę na stanowisku Junior DevOps Engineer lub efektywną współpracę w zespołach deweloperskich stosujących metodologię DevOps

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Rozumie zasady i filozofię DevOps oraz rolę DevOps Engineera w organizacji.	Uczestnik formułuje definicję DevOps, wymienia co najmniej 3 korzyści z jej wdrożenia i omawia różnice między modelem waterfall a podejściem DevOps	Test teoretyczny z wynikiem generowanym automatycznie
Konfiguruje repozytorium Git i stosuje technikę Gitflow w pracy zespołowej	Uczestnik samodzielnie tworzy repozytorium, wykonuje commit, zakłada nową gałąź feature, scala ją z gałęzią develop i poprawnie rozwiązuje konflikt scalania.	Test teoretyczny z wynikiem generowanym automatycznie
Tworzy skrypty Bash realizujące wybrane zadania administracyjne.	Uczestnik pisze skrypt zawierający instrukcje warunkowe i obsługę błędów, który poprawnie wykonuje określone zadanie administracyjne.	Test teoretyczny z wynikiem generowanym automatycznie
Stosuje dobre praktyki DevOps: zarządzanie wersjami, automatyzacja, dokumentacja	Uczestnik uzasadnia wybór co najmniej 3 praktyk DevOps (np. CI/CD, IaC, monitoring) w analizowanym scenariuszu projektowym i wskazuje ich wpływ na jakość pracy zespołu.	Test teoretyczny z wynikiem generowanym automatycznie

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. Po co DEVOPS... Kto to DEVOPS?

o Kultura czy funkcja?

o Po co devops?

3. GIT

o Przypomnienie i podstawy

o Branching - dlaczego jest ważny

o Gitflow - bez tego ani rusz - po co z tego korzystamy

4. Skrypty shellowe na przykładzie basha

o Zmienne

o Pętle

o Instrukcje warunkowe - if then else

o Obsługa błędów

o Funkcje

o Operator case

o Interfejs graficzny w skryptach

o Przykładowy skrypt do robienia backupu - rollbacku

5. Ansible - wstęp do Ansible

o Czym jest Ansible

o Sposoby instalacji Ansible

o Praca z modułami uruchamianymi z linii komend

o Tworzenie pierwszych playbook-ów

o Role

o Przygotowanie od zera deploymentu aplikacji - tworzenie prostego playbook'a do instalacji i uruchomienia aplikacji

6. REST API

o Wprowadzenie do REST API

o Metody autoryzacji

o Jak wykorzystywać REST API praktycznie z konsoli

7. 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

8. 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?

9. 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)

10. Security i inne

- o Bezpieczeństwo kontenerów i skanowanie pod kątem bezpieczeństwa.

11. Docker Compose

- o Wprowadzenie i instalacja
- o Konfiguracja i format YAML

12. Kubernetes

- o Podstawy architektury
- ☒ API
- ☒ Opis obiektów
- o Omówienie kubectl i podobieństwa z docker
- ☒ Wejście w POD i kontener
- ☒ Zarządzanie kontekstem
- o Repliki, skalowanie a deployment wersji
- o Eksponowanie usług
- ☒ NodePort
- ☒ ClusterIP
- ☒ Ingress
- o Konfiguracja i współdzielenie informacji
- ☒ Sekrety
- ☒ ConfigMaps

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

14. Logowanie

- o Centralne logowanie na przykładzie rsyslog
- o Centralne logowanie z wykorzystaniem Elastic Stack

15. Monitoring

- o Dobre praktyki monitoringu serwerów jak i aplikacji
- o Przegląd aplikacji do monitoringu
- o Instalacja oraz konfiguracja Zabbixa
- o Instalacja oraz konfiguracja Prometheusa

o Grafana - one dashboard to rule them all

16. Konceptje pracy z wieloma środowiskami

o dev, test, QA, UAT, PROD - jak to wszystko połączyć i Utrzymać!

17. Dobre praktyki - 12 factor app

o Przegląd dobrych praktyk przy tworzeniu aplikacji

o Przykłady złych aplikacji

18. CI/CD

o Jak robić Code Review mikroserwisów

o Ułożenie Continuous Integration pipeline - na przykładzie Jenkinsa

Harmonogram

Liczba pozycji harmonogramu: 16

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 16 Wprowadzenie	Maciej Krajewski	08-06-2026	09:00	12:30	03:30
2 z 16 Przerwa Obiadowa	Maciej Krajewski	08-06-2026	12:30	13:00	00:30
3 z 16 Po co DEVOPS... Kto to DEVOPS?/GIT	Maciej Krajewski	08-06-2026	13:00	16:00	03:00
4 z 16 Skrypty shellowe na przykładzie basha/Ansible - wstęp do Ansible	Maciej Krajewski	09-06-2026	09:00	12:30	03:30
5 z 16 Przerwa Obiadowa	Maciej Krajewski	09-06-2026	12:30	13:00	00:30
6 z 16 REST API/Docker – co to jest, gdzie pływa i gdzie dokuje?	Maciej Krajewski	09-06-2026	13:00	16:00	03:00

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
7 z 16 Docker - tworzenie własnych obrazów/Registry - czyli repozytorium obrazów	Maciej Krajewski	10-06-2026	09:00	12:30	03:30
8 z 16 Przerwa Obiadowa	Maciej Krajewski	10-06-2026	12:30	13:00	00:30
9 z 16 Security i inne/ Docker Compose	Maciej Krajewski	10-06-2026	13:00	16:00	03:00
10 z 16 Kubernetes/ Portainer – graficzny interfejs użytkownika - co to jest i czy warto	Maciej Krajewski	11-06-2026	09:00	12:30	03:30
11 z 16 Przerwa Obiadowa	Maciej Krajewski	11-06-2026	12:30	13:00	00:30
12 z 16 Logowanie/Monitoring	Maciej Krajewski	11-06-2026	13:00	16:00	03:00
13 z 16 Koncepcje pracy z wieloma środowiskami/Dobre praktyki - 12 factor app/CI/CD	Maciej Krajewski	12-06-2026	09:00	12:30	03:30
14 z 16 Przerwa Obiadowa	Maciej Krajewski	12-06-2026	12:30	13:00	00:30
15 z 16 Zajęcia szkoleniowe – podsumowanie	Maciej Krajewski	12-06-2026	13:00	15:30	02:30
16 z 16 Walidacja	Maciej Krajewski	12-06-2026	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

Podstawowa znajomość systemu Linux, 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

Telefon (+48) 534 506 503