



CI/CD w Jenkins – praktyczne wdrożenia

Numer usługi 2026/02/10/202247/3322933

2 952,00 PLN brutto

2 400,00 PLN netto

210,86 PLN brutto/h

171,43 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
- 🕒 14:00 h
- 📅 05.11.2026 do 06.11.2026

Informacje podstawowe

Kategoria

Informatyka i telekomunikacja / Administracja IT i systemy komputerowe

Grupa docelowa usługi

Skierowane do: Inżynierów DevOps i SysOps, którzy chcą pogłębić wiedzę z zakresu CI/CD w Jenkinsie,

Programistów pragnących zrozumieć pełen cykl budowania i wdrażania aplikacji,

Administratorów systemów i specjalistów IT poszukujących narzędzi do efektywnego zarządzania wydaniem i środowiskami,

Architektów oprogramowania oraz liderów technicznych planujących standaryzację procesu ciągłej integracji i dostarczania w swoich projektach.

Minimalna liczba uczestników

6

Maksymalna liczba uczestników

15

Data zakończenia rekrutacji

02-11-2026

Forma prowadzenia usługi

zdalna w czasie rzeczywistym

Liczba godzin usługi

14

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 praktycznych umiejętności budowania i zarządzania potokami CI/CD z użyciem Jenkins, umożliwiających samodzielne wdrożenie automatycznego procesu integracji i dostarczania oprogramowania w środowisku produkcyjnym lub deweloperskim.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Instaluje i konfiguruje Jenkins oraz zarządza wtyczkami.	Uczestnik instaluje Jenkins, konfiguruje wymagane wtyczki (Git, Pipeline, Docker) i tworzy użytkownika z odpowiednimi uprawnieniami	Test teoretyczny z wynikiem generowanym automatycznie
Tworzy pipeline CI/CD jako kod (Jenkinsfile) z etapami build, test, deploy.	Uczestnik pisze Jenkinsfile z co najmniej 3 etapami (checkout, build, test/deploy), który uruchamia się poprawnie i raportuje wynik każdego etapu.	Test teoretyczny z wynikiem generowanym automatycznie
Integruje Jenkins z systemem kontroli wersji Git i rejestrami kontenerów	Uczestnik konfiguruje wyzwalanie pipeline'u po pushu do repozytorium Git i buduje obraz Docker automatycznie wypchnięty do rejestru po pozytywnym wyniku testów.	Test teoretyczny z wynikiem generowanym automatycznie
Konfiguruje powiadomienia i zarządza artefaktami w Jenkins.	Uczestnik konfiguruje pipeline archiwizujący artefakty buildów i wysyłający powiadomienia e-mail/Slack o wyniku, które działają poprawnie dla scenariuszy sukcesu i błędu.	Test teoretyczny z wynikiem generowanym automatycznie

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 rozdzielanie procesów kształcenia i szkolenia od walidacji?

TAK

Program

Dzień 1 – Fundamenty CI/CD i konfiguracja środowiska

1. Wprowadzenie do CI/CD

1.1. Koncepcje i cele

Definicja Continuous Integration i Continuous Delivery/Deployment

Korzyści biznesowe i techniczne (szybsze release'y, mniej błędów, automatyzacja)

CI/CD a tradycyjny proces wytwarzania oprogramowania

1.2. Popularne narzędzia na rynku

Jenkins vs GitLab CI, GitHub Actions, Azure DevOps

Dlaczego Jenkins pozostaje standardem w dużych organizacjach

2. Architektura i instalacja Jenkinsa pod kątem CI/CD

2.1. Wersje i komponenty

Master/Controller i Agent/Node – podział ról

Jenkins LTS vs Weekly – co wybrać dla CI/CD

2.2. Instalacja i pierwsze uruchomienie

Wersja standalone, Docker Image, Helm Chart (Kubernetes)

Konfiguracja wstępna i użytkownicy

2.3. Konfiguracja pluginów kluczowych dla CI/CD

Pipeline, Git, Blue Ocean, Credentials Binding, Build Pipeline View

Automatyczne aktualizacje i zarządzanie kompatybilnością

3. Podstawy Jenkins Pipeline

3.1. Deklaratywny vs skryptowy Pipeline

Różnice, zalety i wady

3.2. Plik Jenkinsfile

Struktura, etapy, kroki, zmienne środowiskowe

Parametryzacja buildów

3.3. Pierwszy kompletny pipeline

Checkout z Git

Kompilacja/Build

Testy automatyczne

Publikacja artefaktów

4. Integracja z systemami kontroli wersji

4.1. Git/GitHub/GitLab/Bitbucket

Webhooki i automatyczne wyzwalanie buildów

Obsługa branchy i Pull Requestów

4.2. Strategie rozgałęzień (GitFlow, trunk-based)

Mapowanie strategii Git na pipeline Jenkins

Dzień 2 – Zaawansowane potoki i wdrożenia produkcyjne

5. Rozszerzone scenariusze CI

5.1. Testowanie wieloetapowe

Unit test, integration test, smoke test

Raporty JUnit, integracja z SonarQube

5.2. Budowanie i wersjonowanie artefaktów

Maven/Gradle/NPM – przykłady

Publikacja do Nexus/Artifactory/Docker Registry

6. Continuous Delivery & Deployment

6.1. Różnice CD vs CD (Delivery vs Deployment)

6.2. Strategie wdrożeń

Rolling update

Blue-Green

Canary Releases

6.3. Pipeline wdrożeniowy krok po kroku

Środowiska Dev → Test → Stage → Prod

Automatyczne zatwierdzenia (manual input)

Revert i rollback

7. Integracje chmurowe i konteneryzacja

7.1. Docker w potokach

Budowa obrazów, push do registry

Multi-stage builds

7.2. Kubernetes i Helm

Deployment do K8s z Jenkinsa

Jenkins agents jako pody

8. Bezpieczeństwo i zarządzanie tajnymi danymi

8.1. Credentials & Secrets Management

Hasła, klucze SSH, tokeny API

Integracja z HashiCorp Vault / Kubernetes Secrets

8.2. Role-Based Access Control (RBAC)

Uprawnienia dla zespołów Dev, QA, Ops

9. Monitoring i utrzymanie pipeline'ów

9.1. Logi, metryki i alerty

Jenkins Monitoring, Prometheus + Grafana

9.2. Najczęstsze problemy i ich diagnostyka

„stuck builds”, długie kolejki, bottlenecki

9.3. Best practices i automatyczne self-healing

10. Warsztat końcowy – Kompletny pipeline produkcyjny

Stworzenie od zera pełnego przepływu CI/CD:

Commit kodu do Git → wyzwolenie pipeline

Budowa, testy, generowanie artefaktów

Publikacja do Docker Registry

Automatyczne wdrożenie do Kubernetes Stage

Manualne zatwierdzenie i deploy na Production

Harmonogram

Liczba pozycji harmonogramu: 7

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 7 Wprowadzenie do CI/CD/Architektura i instalacja Jenkinsa pod kątem CI/CD/Podstawy Jenkins Pipeline	Piotr Kośka	05-11-2026	09:00	12:30	03:30
2 z 7 Przerwa Obiadowa	Piotr Kośka	05-11-2026	12:30	13:00	00:30
3 z 7 Integracja z systemami kontroli wersji/Rozszerzone scenariusze CI/Continuous Delivery & Deployment	Piotr Kośka	05-11-2026	13:00	16:00	03:00

Przedmiot / temat	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
4 z 7 Integracje chmurowe i konteneryzacja/ Bezpieczeństwo i zarządzanie tajnymi danymi	Piotr Kośka	06-11-2026	09:00	12:30	03:30
5 z 7 Przerwa Obiadowa	Piotr Kośka	06-11-2026	12:30	13:00	00:30
6 z 7 Monitoring i utrzymanie pipeline'ów/Warsztat końcowy – Kompletny pipeline produkcyjny	Piotr Kośka	06-11-2026	13:00	15:30	02:30
7 z 7 Walidacja	Piotr Kośka	06-11-2026	15:30	16:00	00:30

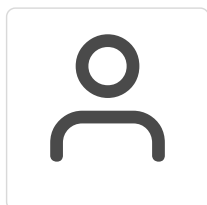
Cennik

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	2 952,00 PLN
Koszt przypadający na 1 uczestnika netto	2 400,00 PLN
Koszt osobogodziny brutto	210,86 PLN
Koszt osobogodziny netto	171,43 PLN

Prowadzący

Liczba prowadzących: 1



1 z 1

Piotr Kośka

Trener posiada wieloletnie doświadczenie w obszarze DevOps i automatyzacji infrastruktury, przy czym kluczowe kwalifikacje w zakresie Terraform, Kubernetes, Jenkins, GitHub Actions oraz wdrożeń chmurowych (AWS) zostały zdobyte i są czynnie wykorzystywane w okresie ostatnich 5 lat

(od 2021 roku do chwili obecnej). Potwierdzają to zrealizowane projekty IaC i CI/CD dla klientów komercyjnych oraz szkolenia z DevOps i Terraform 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, podstawy Git i umiejętność poruszania się po konsoli / terminalu, 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