JSYSTEMS

Wygenerowano: 07.11.2026, 20:38

DEVOPS

Jenkins dla Administratorów i DevOps - CI/CD od podstaw

2 800 PLN
cena netto
(3 444 PLN brutto)

3 dni
czas trwania

Przejdź do strony tego szkolenia →

Dofinansowanie BUR / KFS — nawet do 100%

To szkolenie sfinansujesz ze środków publicznych. Jesteśmy zarejestrowanym dostawcą w Bazie Usług Rozwojowych (BUR); uzupełniamy naszą część wniosku jako realizator.

BUR
do 80%
firmy MŚP i osoby prywatne

KFS
do 100%
dla mikroprzedsiębiorstw

Sprawdź dofinansowanie: jsystems.pl/dofinansowanie →

Najbliższe terminy

Lp.	Data od	Data do	Dni	Termin gwarantowany	Trener	Zapisz się
1	30.11.2026	02.12.2026	3	NIE	Piotr Kośka	Zapisuję się!
2	08.03.2027	10.03.2027	3	NIE	Piotr Kośka	Zapisuję się!
3	07.06.2027	09.06.2027	3	NIE	Piotr Kośka	Zapisuję się!

Prowadzący

Piotr Kośka

Od 2005 roku jest zawodowo związany z branżą IT. Karierę rozpoczął w obszarze wsparcia technicznego (support), a wraz z rozwojem kompetencji przechodził przez kolejne role: System Administratora, DevOps Engineera, Cloud Engineera, Software Engineera, aż po stanowiska liderskie i menedżerskie. Doświadczenie Swoje doświadczenie budował zarówno poprzez intensywną, samodzielną naukę, jak i realizację projektów oraz współpracę z firmami takimi jak Hamas Pay, Tieto, 3Shape, Intel, OVH, Huuuge Games, Sopra Steria, Anixe czy GPW. Współpracował również z wieloma innymi klientami, których nie może ujawnić ze względu na umowy NDA. Specjalizacja Specjalizuje się w obszarze DevOps i Cloud, koncentrując się na automatyzacji procesów, budowie pipeline'ów oraz optymalizacji środowisk IT. Na co dzień pracuje z narzędziami do CI/CD i automatyzacji, takimi jak GitHub Actions, Jenkins, GitLab CI, Azure DevOps, TeamCity czy n8n. W obszarze Infrastructure as Code wykorzystuje rozwiązania takie jak Terraform i Ansible, umożliwiające skalowalne i powtarzalne zarządzanie infrastrukturą. Posiada doświadczenie zarówno w pracy z infrastrukturą on-premise (KVM, Proxmox, VMware, XCP-ng, Hyper-V), jak i rozwiązaniami chmurowymi (AWS, Azure, GCP, DigitalOcean, OVH, Hetzner, Linode i inne). Jako trener W szkoleniach, które prowadzi, stawia przede wszystkim na praktykę. Teorię ogranicza do absolutnego minimum, skupiając się na realnych scenariuszach i zadaniach, które uczestnicy wykonują samodzielnie. Wierzy, że najskuteczniejsza nauka odbywa się poprzez działanie — dlatego jego szkolenia są oparte na pracy z rzeczywistymi problemami, a nie tylko na prezentacjach.

in https://pl.linkedin.com/in/piotrek-s-koska

Opis szkolenia

W skrócie

Trzy dni i te same polecenia, które dziś klikasz ręcznie na kolejnych serwerach, wykonuje za Ciebie jeden powtarzalny proces - z pełną historią tego, kto, co i kiedy uruchomił.

Jenkins to najpopularniejsza platforma do budowania procesów CI/CD (ciągła integracja i dostarczanie zmian), czyli automatycznych potoków, które przejmują ręczną, powtarzalną pracę administratora: wdrożenia, aktualizacje, uruchamianie skryptów, zmiany konfiguracji. Poznajesz Jenkins z dwóch stron naraz - jako osoba budująca potoki i jako administrator odpowiedzialny za instalację, bezpieczeństwo i utrzymanie całej platformy.

Zaczynamy od instalacji i architektury (controller, czyli serwer sterujący, oraz agenci wykonujący zadania), przez pierwsze zadania i integrację z Git, aż po wieloetapowe potoki jako kod, które same walidują zmianę, pobierają sekrety, uruchamiają Terraform i Ansible, a na produkcję wdrażają dopiero po ręcznym zatwierdzeniu.

Co potrafisz po tych 3 dniach

- Instalujesz i konfigurujesz Jenkins od zera - stawiasz platformę na Linuksie, ustawiasz administratora, użytkowników i uprawnienia, aktualizujesz wtyczki bez rozsypania środowiska.
- Rozumiesz architekturę controller/agent - wiesz, czemu ciężkie zadania nie mogą lądować na serwerze sterującym, i kierujesz je na właściwych agentów Linux albo Windows.
- Piszesz potoki jako kod (Jenkinsfile) - budujesz wieloetapowe procesy trzymane razem z kodem w Git, z parametrami oraz krokami warunkowymi i równoległymi.
- Bezpiecznie zarządzasz sekretami - hasła, tokeny i klucze SSH trzymasz w magazynie Credentials, używasz ich w potoku i pilnujesz, żeby nie wyciekły do logów.
- Uruchamiasz Terraform i Ansible z poziomu Jenkinsa - zmiana infrastruktury zaczyna się od zmiany kodu w Git, jest automatycznie walidowana, a wdrażana dopiero po akceptacji.

• Wdrażasz na produkcję pod kontrolą - budujesz osobne procesy dla środowisk DEV, TEST i PROD, z ręcznym zatwierdzaniem operacji (mechanizm input) i obsługą błędów, żeby żadna niebezpieczna zmiana nie poszła przypadkiem.
• Diagnozujesz zepsute potoki - czytasz Console Output i logi, rozpoznajesz brak wolnego agenta, błędne sekrety, problemy z Git czy uprawnieniami i wiesz, od czego zacząć.
• Utrzymujesz platformę w ryzach - robisz kopię zapasową katalogu JENKINS_HOME i odtwarzasz środowisko, czyścisz workspace, panujesz nad miejscem na dysku i podstawami monitorowania.

Jak pracujemy

Uczymy w praktyce. Każdy z trzech dni zamyka warsztat (LAB) wykonywany na środowisku Jenkins, a całość spina projekt końcowy: kompletny potok, który idzie od zmiany w Git, przez walidację i narzędzia infrastruktury (Terraform, Ansible), po zatwierdzenie i wdrożenie na produkcję z weryfikacją wyniku. Potoki, które budujesz, nie służą tylko do uruchamiania buildów - wykonują realne zadania administracyjne: restart usług, aktualizacje serwerów, wdrażanie konfiguracji, zadania z harmonogramu. Pracujesz na agentach Linux (polecenia Bash) i Windows (PowerShell). Prowadzi Piotr Kośka, specjalista DevOps i Cloud skupiony na budowie pipeline'ów, który na co dzień pracuje z narzędziami CI/CD, takimi jak Jenkins, GitLab CI i GitHub Actions.

Dla kogo

Szkolenie jest dla administratorów systemów Linux i Windows, inżynierów DevOps i Cloud, osób odpowiedzialnych za automatyzację IT oraz zespołów, które chcą przenieść ręcznie wykonywane operacje do kontrolowanych potoków CI/CD. To także dobry punkt startu dla osób, które z Jenkinsem nie pracowały wcześniej.

Wymagania wstępne: podstawowa znajomość administracji Linuksem lub Windowsem oraz podstaw Git jest pomocna, ale samego Jenkinsa poznajesz od zera - wcześniejsze doświadczenie z nim nie jest potrzebne.

Co zyskujesz

Wychodzisz z działającym potokiem CI/CD zbudowanym własnymi rękami i z wiedzą, jak samodzielnie zaprojektować, uruchomić i utrzymać taki proces we własnej infrastrukturze. Otrzymujesz certyfikat ukończenia szkolenia, a udział można sfinansować ze środków BUR i KFS. Szkolenie realizujemy również w formule zamkniętej dla zespołu.

Zarezerwuj miejsce i zamień ręczne wdrożenia na proces, który wykonuje się sam i zostawia po sobie ślad.

Potrzebujesz pełnego obrazu DevOps? Zobacz kompleksowe szkolenie DevOps - 5 dni warsztatów: CI/CD, konteneryzacja, IaC i automatyzacja w jednym.

Dofinansowanie KFS/BUR

Sprawdź, czy w Twoim przypadku dostępne jest finansowanie szkolenia ze środków BUR/KFS. Dostępność finansowania zależy od warunków danego programu oraz decyzji właściwej instytucji. Więcej informacji: https://jsystems.pl/dofinansowanie/

Program szkolenia

1. Wprowadzenie do CI/CD

• czym jest Continuous Integration,
• czym jest Continuous Delivery i Continuous Deployment,
• rola Jenkins w procesach CI/CD,
• CI/CD z perspektywy administratora systemów,

- automatyzacja procesów infrastrukturalnych,
- Jenkins a GitHub Actions, GitLab CI/CD i Bitbucket Pipelines,
- kiedy warto utrzymywać własną platformę Jenkins,
- przykładowe zastosowania Jenkins w środowiskach administracyjnych i DevOps.

2. Architektura Jenkins

- główne komponenty Jenkins,
- Jenkins Controller,
- Jenkins Agent,
- Executor,
- Job,
- Workspace,
- Pipeline,
- Node,
- Plugin,
- Queue,
- sposób wykonywania zadań przez Jenkins,
- komunikacja pomiędzy controllerem i agentami,
- planowanie infrastruktury Jenkins.

3. Instalacja Jenkins

- wymagania systemowe,
- instalacja Java,
- instalacja Jenkins na Linux,
- struktura katalogów Jenkins,
- katalog JENKINS_HOME,
- pierwsze uruchomienie,
- konfiguracja administratora,
- konfiguracja URL Jenkins,
- podstawowe parametry systemowe.

4. Zarządzanie Jenkins

- konfiguracja globalna,
- zarządzanie użytkownikami,
- zarządzanie pluginami,
- instalacja i aktualizacja pluginów,
- wpływ pluginów na stabilność Jenkins,
- restart i bezpieczny restart Jenkins,
- logi systemowe,
- podstawowa diagnostyka platformy.

5. Bezpieczeństwo Jenkins

- uwierzytelnianie użytkowników,
- autoryzacja,
- Role-Based Access Control,
- Matrix Authorization,
- zasada najmniejszych uprawnień,
- zabezpieczenie dostępu do Jenkins,

- podstawowe dobre praktyki bezpieczeństwa,
- ograniczanie dostępu do administracji platformą.

6. Pierwsze zadania Jenkins

- tworzenie pierwszego Job,
- Freestyle Project,
- konfiguracja kroków wykonywania,
- uruchamianie poleceń Bash,
- uruchamianie PowerShell,
- zmienne środowiskowe Jenkins,
- parametryzacja zadań,
- historia wykonań,
- analiza Console Output,
- obsługa kodów zakończenia procesów.

7. Integracja Jenkins z Git

- podstawy integracji Git i Jenkins,
- konfiguracja repozytorium,
- klonowanie kodu przez Jenkins,
- HTTPS oraz SSH,
- credentials do repozytoriów,
- webhooki,
- automatyczne uruchamianie pipeline po zmianie kodu,
- praca z branchami.

LAB

- Uczestnicy instalują własne środowisko Jenkins, wykonują jego podstawową konfigurację oraz tworzą pierwsze zadania automatyzujące operacje administracyjne.

8. Wprowadzenie do Jenkins Pipeline

- czym jest Pipeline as Code,
- różnice pomiędzy Freestyle Job i Pipeline,
- Declarative Pipeline,
- Scripted Pipeline,
- struktura pipeline,
- podstawowa składnia.

9. Jenkinsfile

- czym jest Jenkinsfile,
- przechowywanie pipeline wraz z kodem,
- wersjonowanie pipeline w Git,
- podstawowa struktura Jenkinsfile,
- sekcja pipeline,
- agent,
- stages,
- stage,
- steps,
- post,

- environment,
- parameters.

10. Budowa wieloetapowych pipeline'ów

Przykładowy proces:

- pobranie kodu,
- walidacja,
- wykonanie testów,
- przygotowanie środowiska,
- deployment,
- weryfikacja,
- publikacja wyników.

Omówione zostaną również:

- zależności pomiędzy etapami,
- zatrzymywanie pipeline po błędzie,
- obsługa kodów wyjścia,
- wykonywanie kroków warunkowych,
- wykonywanie kroków równolegle.

11. Parametryzacja pipeline

- parametry tekstowe,
- wartości logiczne,
- listy wyboru,
- parametry środowiska,
- dynamiczne sterowanie pipeline,
- wybór środowiska DEV, TEST i PROD,
- wykorzystywanie parametrów w kolejnych etapach.

12. Zmienne środowiskowe

- globalne zmienne Jenkins,
- zmienne pipeline,
- zmienne poszczególnych stage,
- wykorzystanie zmiennych w Bash i PowerShell,
- przekazywanie wartości pomiędzy etapami.

13. Credentials i zarządzanie sekretami

- Jenkins Credentials Store,
- Username/Password,
- Secret Text,
- SSH Private Key,
- Token,
- wykorzystanie credentials w pipeline,
- withCredentials,
- ochrona sekretów w logach,
- najczęstsze błędy związane z sekretami,
- dobre praktyki zarządzania danymi uwierzytelniającymi.

14. Warunki i kontrola wykonywania pipeline

- sekcja when,
- wykonywanie etapów dla wybranych branchy,
- różne procesy dla DEV, TEST i PROD,
- ręczne zatwierdzanie operacji,
- mechanizm input,
- kontrola wdrożeń produkcyjnych.

15. Obsługa błędów

- interpretacja statusu pipeline,
- analiza Console Output,
- post success,
- post failure,
- post always,
- timeout,
- retry,
- reakcja pipeline na błędy,
- kontrolowane przerywanie wykonywania.

16. Artefakty i wyniki wykonywania

- archiwizacja plików,
- przekazywanie artefaktów,
- raporty,
- logi,
- przechowywanie wyników wykonywania,
- wykorzystanie artefaktów pomiędzy etapami.

LAB

Budowa kompletnego Jenkinsfile, który:

- pobiera kod z Git,
- wykonuje walidację,
- przyjmuje parametry,
- wykorzystuje credentials,
- wykonuje kilka etapów,
- reaguje na błędy,
- wymaga zatwierdzenia przed wdrożeniem na środowisko produkcyjne.

17. Jenkins Controller i Agent

- dlaczego nie należy wykonywać wszystkich zadań na controllerze,
- budowa środowiska wieloagentowego,
- instalacja i konfiguracja agentów,
- statyczne i dynamiczne agenty,
- labels,
- executors,
- kierowanie pipeline na odpowiedniego agenta,
- różne agenty dla różnych technologii.

18. Pipeline'y dla Linux i Windows

- agent Linux,
- agent Windows,
- wykonywanie Bash,
- wykonywanie PowerShell,
- wybór agenta na podstawie label,
- budowa pipeline pracującego na różnych systemach operacyjnych.

19. Jenkins i Ansible

- przygotowanie agenta Jenkins dla Ansible,
- przechowywanie playbooków w Git,
- uruchamianie ansible-playbook,
- przekazywanie parametrów,
- obsługa inventory,
- credentials SSH,
- Ansible Vault,
- automatyzacja zmian konfiguracyjnych.

20. Jenkins i Terraform

- przygotowanie agenta Jenkins dla Terraform,
- terraform fmt,
- terraform validate,
- terraform init,
- terraform plan,
- terraform apply,
- obsługa różnych środowisk,
- zabezpieczenie wykonywania apply,
- ręczna akceptacja zmian infrastruktury,
- credentials do platform chmurowych,
- podstawowe podejście do pracy z Terraform State.

21. Automatyzacja zadań administracyjnych

Przykłady zastosowania Jenkins do:

- wykonywania skryptów Bash,
- wykonywania PowerShell,
- restartowania usług,
- wdrażania konfiguracji,
- aktualizacji serwerów,
- wykonywania kontroli środowiska,
- automatyzacji operacji utrzymaniowych,
- uruchamiania zadań według harmonogramu.

22. Triggery i automatyczne uruchamianie pipeline

- ręczne uruchamianie,
- webhook,
- Git push,
- harmonogram,
- cron,
- uruchamianie pipeline przez inny pipeline,

- podstawy Jenkins API.

23. Organizacja Jenkins w większym środowisku

- foldery,
- konwencje nazewnicze,
- podział projektów,
- organizacja pipeline'ów,
- oddzielanie DEV, TEST i PROD,
- współdzielenie konfiguracji,
- podstawy Shared Libraries,
- ograniczanie duplikowania Jenkinsfile.

24. Utrzymanie Jenkins

- aktualizacje Jenkins,
- aktualizacje pluginów,
- kopia zapasowa JENKINS_HOME,
- odtwarzanie środowiska,
- zarządzanie historią buildów,
- czyszczenie workspace,
- wykorzystanie przestrzeni dyskowej,
- podstawowe aspekty monitorowania Jenkins.

25. Diagnostyka problemów

- pipeline pozostający w kolejce,
- brak dostępnego executora,
- niedostępny agent,
- problemy z Git,
- błędne credentials,
- problemy z pluginami,
- błędy Jenkinsfile,
- problemy z uprawnieniami,
- analiza Console Output,
- analiza logów Jenkins,
- metodyczne podejście do troubleshooting.

26. Dobre praktyki Jenkins

- Pipeline as Code,
- przechowywanie Jenkinsfile w Git,
- minimalizacja operacji wykonywanych na controllerze,
- separacja agentów,
- bezpieczne zarządzanie sekretami,
- ograniczanie uprawnień,
- kontrolowane deploymenty produkcyjne,
- wersjonowanie konfiguracji,
- ograniczenie zależności od pluginów,
- budowanie powtarzalnych procesów automatyzacji.

LAB końcowy

Uczestnicy budują kompletny pipeline automatyzujący proces infrastrukturalny:

Git - Jenkins - walidacja - Terraform/Ansible - akceptacja - wdrożenie - weryfikacja

Pipeline wykorzystuje:

- repozytorium Git,
- Jenkinsfile,
- parametry,
- credentials,
- agenta Jenkins,
- kilka stage'y,
- obsługę błędów,
- zatwierdzenie operacji produkcyjnej,
- narzędzia automatyzacji infrastruktury.

Opinie uczestników

★★★★★
„Świetnie”
Robert Karolewski — Bank Gospodarstwa Krajowego

★★★★★
„Polecam!”
Karol Bury — Asseco Poland S.A.

★★★★★
„Cacy :)”
Mateusz Filipowicz — Advatech sp. z o.o

...i wiele więcej opinii o tym szkoleniu →

Google

Opinie naszych kursantów

5/5

★★★★★

na podstawie 302 opinii w Google

Zobacz opinie w Google →

Szkolenie dedykowane dla Twojego zespołu

Można je zorganizować w formie zamkniętej — z programem dopasowanym do potrzeb Twojego zespołu, w uzgodnionym terminie i formacie.

Przejdź do strony tego szkolenia →

Dlaczego warto nas wybrać?

✔

Terminy gwarantowane na 100%

Jeśli termin jest oznaczony jako gwarantowany, odbędzie się we wskazanym czasie — nawet jeśli część grupy wycofa się z udziału. Ryzyko organizacyjne ponosimy my, nie Ty.

★

Trenerzy-praktycy z oceną minimum 4.75/5

Każdy trener JSystems ma bogate doświadczenie komercyjne w technologiach których uczy i musi utrzymywać średnią z ankiet powyżej 4.75 na 5. Czerwony pasek to u nas minimum ;)

⚙

100% format warsztatowy

Każde zagadnienie teoretyczne jest poparte ćwiczeniami praktycznymi o rosnącym poziomie trudności. Uczymy przez działanie, nie przez slajdy.

%

Jesteśmy w BUR

Jako zarejestrowany dostawca usług w Bazie Usług Rozwojowych umożliwiamy sfinansowanie szkoleń pracowników ze środków zewnętrznych — dofinansowanie z BUR i KFS sięga nawet 100% wartości szkolenia.

✪

Certyfikat ISO 9001

Posiadamy certyfikat ISO 9001 potwierdzający najwyższe standardy jakości naszych usług szkoleniowych i procesów organizacyjnych.

☁

Gotowe środowiska w chmurze

Do każdego szkolenia wymagającego oprogramowania dostarczamy skonfigurowane, gotowe hosty w chmurze. Uczestnik nie musi nic instalować — łączy się przez zdalny pulpit lub SSH.

Zainteresowało Cię to szkolenie?

Zapisz się online lub skontaktuj się z nami — pomożemy dobrać odpowiedni termin.

Przejdź do zapisu →

Zaufali nam

SAMSUNG · Volkswagen · orange · PLAY · ING BANK

...oraz setki innych firm

Kontakt

JSystems Sp. z o.o.

ul. Żelazna 59A lok. 6.1, 00-848 Warszawa

NIP: 525 259 75 20 | KRS: 0000528203 | REGON: 147470200

☎ 22 299 53 69

✉ biuro@jsystems.pl

➤ www.jsystems.pl