



Ernabo Adrian Flak

★★★★★ 4,6 / 5

767 ocen

## Szkolenie: System kontroli wersji GIT – Podstawy i Praktyka

Numer usługi 2025/11/25/22948/3169944

📍 zdalna w czasie rzeczywistym

🏠 Usługa szkoleniowa

🕒 31 h

📅 02.03.2026 do 30.03.2026

3 720,00 PLN brutto

3 720,00 PLN netto

120,00 PLN brutto/h

120,00 PLN netto/h

## Informacje podstawowe

### Kategoria

Informatyka i telekomunikacja / Administracja IT i systemy komputerowe

### Identyfikatory projektów

Kierunek - Rozwój, Małopolski Pociąg do kariery, Zachodniopomorskie Bony Szkoleniowe

- **Programiści i testerzy** – chcący poznać i wykorzystać GIT w pracy zespołowej
- **Administratorzy systemów** – wdrażanie repozytoriów i kontroli wersji
- **Zespoły deweloperskie** – automatyzacja przepływów pracy i współpraca w repozytoriach

### Wymagania wstępne:

- Podstawowa znajomość systemu operacyjnego i terminala/CLI
- Podstawowa wiedza programistyczna

### Grupa docelowa usługi

- Szkolenie przeznaczone jest również dla uczestników projektu **Kierunek Rozwój** realizowany przez WUP w Toruniu.
- Usługa również adresowana dla Uczestników Projektu **Małopolski Pociąg do Kariery sezon 1**
- Usługa skierowana również dla uczestników projektu "Zachodniopomorskie bony szkoleniowe"
- Oraz dla uczestników projektów dofinansowanych w całej Polsce
- Szkolenie skierowane jest zarówno do **osób indywidualnych, jak i pracodawców i ich pracowników.**

### Minimalna liczba uczestników

5

### Maksymalna liczba uczestników

10

### Data zakończenia rekrutacji

26-02-2026

### Forma prowadzenia usługi

zdalna w czasie rzeczywistym

# Cel

## Cel edukacyjny

Usługa przygotowuje uczestnika do wykorzystania systemu kontroli wersji GIT – od podstawowych operacji, śledzenia zmian i pracy na gałęziach, po zarządzanie repozytoriami zdalnymi, rozwiązywanie konfliktów, tagowanie oraz automatyzację workflow i integrację z CI/CD. Zdobytą wiedzę uczestnik zastosuje w realizacji kompletnego projektu zespołowego.

## Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

| Efekty uczenia się                                                                | Kryteria weryfikacji                                                                   | Metoda walidacji                                      |
|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|-------------------------------------------------------|
| Uczestnik identyfikuje funkcje i strukturę systemu kontroli wersji GIT.           | Rozpoznaje podstawowe polecenia GIT (init, status, log) w zadaniu wielokrotnego wyboru | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                                   | Wskazuje elementy repozytorium lokalnego i zdalnego w scenariuszu testowym             | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                                   | Konfiguruje podstawowe ustawienia GIT w środowisku praktycznym                         | Test teoretyczny z wynikiem generowanym automatycznie |
| Uczestnik śledzi i zarządza zmianami w plikach projektu.                          | Dodaje i zatwierdza zmiany przy użyciu git add i git commit                            | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                                   | Cofnięcie zmian lokalnych w praktycznym zadaniu (git reset, git checkout).             | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                                   | Przegląda historię zmian w repozytorium (git log)                                      | Test teoretyczny z wynikiem generowanym automatycznie |
| Uczestnik tworzy i zarządza gałęziami (branching) oraz wykonuje scalanie (merge). | Tworzy nowe gałęzie i przełącza się między nimi (git branch, git checkout)             | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                                   | Scala gałęzie i rozwiązuje konflikty w scenariuszu praktycznym (git merge)             | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                                   | Praktycznie stosuje rebase i strategię pracy zespołowej z branchami                    | Test teoretyczny z wynikiem generowanym automatycznie |

| Efekty uczenia się                                                  | Kryteria weryfikacji                                                                  | Metoda walidacji                                      |
|---------------------------------------------------------------------|---------------------------------------------------------------------------------------|-------------------------------------------------------|
| Uczestnik zarządza repozytoriami zdalnymi i współpracą zespołową.   | Tworzy repozytorium zdalne i łączy je z lokalnym (git remote add, git push)           | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                     | Pobiera zmiany z repozytorium zdalnego (git pull, git fetch)                          | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                     | Rozwiązuje konflikty powstałe podczas synchronizacji zdalnej                          | Test teoretyczny z wynikiem generowanym automatycznie |
| Uczestnik stosuje zaawansowane funkcje GIT i automatyzuje workflow. | Tworzy tagi i zarządza wersjonowaniem projektu (git tag)                              | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                     | Konfiguruje podstawowe hooki GIT (pre-commit, post-commit) w praktyce online          | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                     | Integruje repozytorium z procesami CI/CD w zadaniu praktycznym                        | Test teoretyczny z wynikiem generowanym automatycznie |
| Uczestnik tworzy kompletny projekt zespołowy z wykorzystaniem GIT.  | Zakłada repozytorium lokalne i zdalne oraz wykonuje branching w projekcie praktycznym | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                     | Wspólnie z zespołem scala zmiany, rozwiązuje konflikty i wykonuje commit/push         | Test teoretyczny z wynikiem generowanym automatycznie |
|                                                                     | Prezentuje projekt końcowy i weryfikuje poprawność wszystkich operacji GIT            | 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?

# Program

Program szkolenia jest dostosowany do potrzeb uczestników usługi oraz głównego celu usługi i jej charakteru oraz obejmuje zakres tematyczny usługi. Uczestnik nie musi spełniać dodatkowych wymagań dot. poziomu zaawansowania.

**Usługa prowadzona jest w godzinach dydaktycznych. Przerwy nie są wliczone w ogólny czas usługi rozwojowej.** Harmonogram usługi może ulec nieznacznemu przesunięciu, ponieważ ilość przerw oraz długość ich trwania zostanie dostosowana indywidualnie do potrzeb uczestników szkolenia. Łączna długość przerw podczas szkolenia nie będzie dłuższa aniżeli zawarta w harmonogramie.

**Zajęcia zostaną przeprowadzone przez ekspertów z wieloletnim doświadczeniem, którzy przekazuje nie tylko wiedzę teoretyczną, ale także praktyczne wskazówki i najlepsze praktyki. Uczestnicy mają możliwość czerpania z jego wiedzy i doświadczeń.**

Szkolenie będzie realizowane **zdalnie w czasie rzeczywistym** za pomocą platformy **ClickMeeting**, co umożliwia aktywny udział uczestników w warsztatach i ćwiczeniach grupowych.

Szkolenie realizowane jest przez platformę umożliwiającą:

- udostępnianie ekranu,
- czat, komunikację audio-wideo,
- współdzielenie materiałów i plików,
- interaktywną prezentację kodu i analiz danych.

Każdy uczestnik pracuje indywidualnie na swoim komputerze z bieżącym wsparciem trenera.

---

Przed dokonaniem zapisu i złożeniem karty uczestnictwa do Operatora, zachęcamy do **kontaktowania się z nami telefonicznie, SMS-em lub e-mailem** pod adresem/numerem wskazanym w zakładce „**Kontakt**”.

Pozwoli to **potwierdzić dostępność miejsca** w grupie szkoleniowej oraz rozwiązać ewentualne wątpliwości.

---

## Program szkolenia:

### MODUŁ 1. Wprowadzenie do systemu kontroli wersji

**3h dydaktyczne (1h teoria / 2h praktyka)**

#### Teoria:

- Co to jest system kontroli wersji
- Historia GIT i jego zalety
- Typy systemów kontroli wersji

#### Praktyka:

- Instalacja GIT i konfiguracja podstawowa
- Tworzenie pierwszego repozytorium lokalnego
- Podstawowe polecenia

### MODUŁ 2. Podstawowe operacje w GIT

**5h dydaktycznych (2h teoria / 3h praktyka)**

#### Teoria:

- Śledzenie zmian w plikach
- Commit, add, remove
- Koncept stage, commit i history

#### Praktyka:

- Dodawanie i zatwierdzanie zmian
- Przegląd historii zmian
- Cofanie zmian (git reset, git checkout)

### **MODUŁ 3. Gałęzie (branching) i scalanie (merge)**

#### **6h dydaktycznych (2h teoria / 4h praktyka)**

##### **Teoria:**

- Tworzenie i zarządzanie gałęziami
- Merge, rebase, konflikty
- Strategie pracy zespołowej z branchami

##### **Praktyka:**

- Tworzenie gałęzi i przełączanie między nimi
- Scalanie gałęzi (git merge)
- Rozwiązywanie konfliktów w praktyce

### **MODUŁ 4. Repozytoria zdalne i współpraca zespołowa**

#### **5h dydaktycznych (2h teoria / 3h praktyka)**

##### **Teoria:**

- Koncept repozytorium zdalnego
- GitHub, GitLab, Bitbucket
- Pull, push, fetch, clone

##### **Praktyka:**

- Tworzenie repozytorium zdalnego i połączenie z lokalnym
- Wysyłanie zmian do repozytorium zdalnego
- Pobieranie zmian i rozwiązywanie konfliktów

### **MODUŁ 5. Automatyzacja i zaawansowane funkcje GIT**

#### **6h dydaktycznych (2h teoria / 4h praktyka)**

##### **Teoria:**

- Tagowanie i wersjonowanie
- Hooki GIT, aliasy i skróty
- Automatyzacja workflow CI/CD

##### **Praktyka:**

- Tworzenie tagów i wersji w repozytorium
- Konfiguracja podstawowych hooków (pre-commit, post-commit)
- Praktyczne ćwiczenia z integracją workflow

### **MODUŁ 6. Projekt końcowy / warsztaty**

#### **5h dydaktycznych (praktyka)**

- Wspólne stworzenie projektu zespołowego:
  - Tworzenie repozytorium lokalnego i zdalnego
  - Branching i merge
  - Zarządzanie zmianami i rozwiązywanie konfliktów
  - Wdrożenie workflow CI/CD

##### **Walidacja 1h**

# Harmonogram

Liczba przedmiotów/zajęć: 0

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

# Cennik

## Cennik

| Rodzaj ceny                               | Cena         |
|-------------------------------------------|--------------|
| Koszt przypadający na 1 uczestnika brutto | 3 720,00 PLN |
| Koszt przypadający na 1 uczestnika netto  | 3 720,00 PLN |
| Koszt osobogodziny brutto                 | 120,00 PLN   |
| Koszt osobogodziny netto                  | 120,00 PLN   |

# Prowadzący

Liczba prowadzących: 0

Brak wyników.

# Informacje dodatkowe

## Informacje o materiałach dla uczestników usługi

Prezentacja w formacie pdf dla uczestnika po zakończonym kursie.

## Warunki uczestnictwa

Warunkiem zdobycia certyfikatu potwierdzającego zdobyte kompetencje jest przystąpienie do Egzaminu . Na egzamin uczestnik nie musi dokonywać osobnego zapisu oraz jest w koszt usługi.

**Wymagana jest obecność min 80% lub zgodna ze wskazaniami Operatora. Obecność na usłudze weryfikowana będzie na podstawie raportu logowań wygenerowanego z platformy.**

*Uczestnicy przyjmują do wiadomości, że usługa może być poddana monitoringowi z ramienia Operatora lub PARP i wyrażają na to zgodę.*

*Uczestnik ma obowiązek zapisania się na usługę przez BUR co najmniej w dniu zakończenia rekrutacji.*

Organizator zapewnia dostępność osobom ze szczególnymi potrzebami podczas realizacji usług rozwojowych zgodnie z Ustawą z dnia 19 lipca 2019 r. o zapewnianiu dostępności osobom ze szczególnymi potrzebami (Dz.U. 2022 poz. 2240) oraz „Standardami dostępności dla polityki spójności 2021-2027”. **W przypadku potrzeby zapewnienia specjalnych udogodnień prosimy o kontakt przed zapisem na usługę!**

## Informacje dodatkowe

- **Zapis BUR nie jest jednoznaczny z zarezerwowaniem miejsca.** W celu potwierdzenia miejsca prosimy o dodatkowy kontakt telefoniczny/sms lub mailowy na adres/numer wskazany w zakładce "kontakt"
- zawarto umowę z WUP w Toruniu w ramach projektu Kierunek Rozwój
- zawarto umowę z WUP w Krakowie w ramach projektu Małopolski Pociąg do Kariery
- zawarto umowę z WUP w Szczecinie w ramach projektu Zachodniopomorskie Bony Szkoleniowe
- usługi dedykowane również uczestnikom innych programów dofinansowań

### **Podstawa zwolnienia z VAT:**

**1) art. 43 ust. 1 pkt 29 lit. c Ustawy z dnia 11 marca 2024 o podatku od towarów i usług - w przypadku dofinansowania w wysokości 100%**

**2) § 3 ust. 1 pkt. 14 Rozporządzenia Ministra Finansów z dnia 20 grudnia 2013 r. w sprawie zwolnień od podatku od towarów i usług oraz warunków stosowania tych zwolnień - w przypadku dofinansowania w co najmniej 70%**

**3) W przypadku braku uzyskania dofinansowania lub uzyskania dofinansowania poniżej 70%, do ceny usługi należy doliczyć 23% VAT**

## Warunki techniczne

### **1. Sprzęt uczestnika:**

- **komputer lub laptop** z systemem operacyjnym Windows 10 / 11, macOS lub Linux,
- **procesorem** co najmniej **Intel i5 / Ryzen 5** lub równoważnym,
- **pamięcią RAM: minimum 8 GB** (zalecane 16 GB dla płynnej pracy z dużymi zbiorami danych),
- **wolną przestrzenią dyskową: minimum 10 GB,**
- **stabilne łącze internetowe (min. 10 Mbps)** – w przypadku zajęć zdalnych,
- **aktualna przeglądarka internetową (Chrome, Edge, Firefox),**

### **Obowiązkowe:**

- **Kamera:** Uczestnik powinien posiadać działającą kamerę (wbudowaną w laptop/komputer lub zewnętrzną). Kamera umożliwia aktywny udział w sesjach, prezentację ćwiczeń grupowych oraz interakcję z prowadzącym.
- **Mikrofon:** Niezbędny jest sprawny mikrofon (wbudowany lub zewnętrzny, np. w zestawie słuchawkowym). Umożliwia zadawanie pytań, udział w dyskusjach i ćwiczeniach grupowych.
- Zalecane użycie słuchawek z mikrofonem, aby zredukować echo i poprawić jakość dźwięku.

### **2. Oprogramowanie:**

- Nie jest wymagane wcześniejsze przygotowanie środowiska programistycznego. Wszystkie niezbędne programy, dane i narzędzia zostaną przekazane przez trenera w trakcie trwania szkolenia.

### **3. Łącze internetowe:**

- Minimum 10 Mbps download / 5 Mbps upload
- Stabilne połączenie bez dużych przerw i opóźnień

### **4. Środowisko pracy:**

- Ciche miejsce do pracy i nauki
- Dostęp do powierzchni roboczej umożliwiającej komfortowe używanie komputera
- Możliwość dzielenia ekranu w trakcie sesji praktycznych i konsultacji

### **5. Środowisko szkoleniowe**

Szkolenie realizowane jest przez platformę umożliwiającą:

- udostępnianie ekranu,
- czat, komunikację audio-wideo,
- współdzielenie materiałów i plików,
- interaktywną prezentację kodu i analiz danych.

Każdy uczestnik pracuje indywidualnie na swoim komputerze z bieżącym wsparciem trenera.

## Kontakt



**Nikol Watola**

**E-mail** [nikol.watola@ernabo.com](mailto:nikol.watola@ernabo.com)

**Telefon** (+48) 530 642 270