



AI w pracy programisty – Copilot, Claude i ChatGPT w codziennym kodowaniu - szkolenie

Numer usługi 2026/09/28/199788/3841499

1 845,00 PLN brutto
1 500,00 PLN netto
115,31 PLN brutto/h
93,75 PLN netto/h
157,50 PLN cena rynkowa ⓘ

EXPERT-SALES

SPÓŁKA Z

OGRANICZONĄ

ODPOWIEDZIALNOŚĆ

CIĄ

★★★★★ 4,9 / 5

25 ocen

🏠 Usługa szkoleniowa

📄 zdalna w czasie rzeczywistym

👥 Zajęcia grupowe

🕒 16:00 h

📅 21.11.2026 do 22.11.2026

Informacje podstawowe

Kategoria

Informatyka i telekomunikacja / Programowanie

Grupa docelowa usługi

Szkolenie jest skierowane do osób, które zawodowo piszą lub utrzymują kod i chcą świadomie dobrać narzędzia AI do codziennych zadań programistycznych. W szczególności do:

- programistów aplikacji webowych, desktopowych, mobilnych i usług backendowych
- testerów automatyzujących i inżynierów DevOps
- liderów technicznych, którzy wybierają narzędzia AI dla zespołu
- programistów, którzy korzystali z AI okazjonalnie i chcą robić to systematycznie
- studentów i absolwentów kierunków informatycznych, którzy już programują

Wymagana jest praktyczna umiejętność programowania w dowolnym języku oraz podstawy pracy z systemem kontroli wersji git. Szkolenie nie uczy programowania od zera.

Minimalna liczba uczestników

3

Maksymalna liczba uczestników

25

Data zakończenia rekrutacji

05-11-2026

Forma prowadzenia usługi

zdalna w czasie rzeczywistym

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

Szkolenie przygotowuje do samodzielnego i bezpiecznego wykorzystywania narzędzi sztucznej inteligencji (asystentów w edytorze kodu, czatów AI oraz agentów kodujących) w procesie wytwarzania oprogramowania, w tym do dobierania właściwego narzędzia do zadań programistycznych, pisania, debugowania, testowania, refaktoryzacji i dokumentowania kodu oraz krytycznej oceny jakości generowanych rozwiązań.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Charakteryzuje narzędzia AI wspomagające proces wytwarzania oprogramowania	Wymienia różnice architektoniczne i funkcjonalne pomiędzy asystentem w edytorze kodu, czatem konwersacyjnym a autonomicznym agentem programistycznym.	Test teoretyczny
	Klasyfikuje zadania programistyczne pod kątem efektywności zastosowania poszczególnych rodzajów modeli AI.	Test teoretyczny
	Omawia techniki zarządzania oknem kontekstowym oraz strukturyzowania promptów technicznych.	Test teoretyczny
Identyfikuje ryzyka techniczne i prawne związane z kodem generowanym przez AI	Wskazuje typowe podatności bezpieczeństwa oraz błędy logiczne specyficzne dla kodu generowanego automatycznie.	Test teoretyczny
	Omawia kwestie zgodności licencyjnej kodu źródłowego oraz ryzyka naruszenia praw autorskich.	Test teoretyczny
	Definiuje zagrożenia wycieku danych poufnych i kodu własnościowego wynikające z przesyłania danych do zewnętrznych modeli.	Test teoretyczny
	Wyjaśnia modele kosztowe (tokeny, subskrypcje) oraz techniczne limity zapytań (rate limits) poszczególnych platform.	Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Obsługuje asystenta kodowania w środowisku programistycznym (IDE)	Konfiguruje wtyczkę asystenta w edytorze kodu zgodnie z wymaganiami technicznymi projektu.	Test teoretyczny
	Steruje generowaniem odpowiedzi w czasie rzeczywistym za pomocą komentarzy i szablonów kodu.	Test teoretyczny
	Modyfikuje istniejący kod przy użyciu wbudowanego czatu i poleceń kontekstowych.	Test teoretyczny
	Stosuje tryb agenta w edytorze do automatycznej edycji powiązanych plików projektu.	Test teoretyczny
	Tworzy instrukcje uwzględniające precyzyjną rolę, kontekst architektoniczny i pożądany format wyjściowy kodu.	Test teoretyczny
Projektuje zapytania techniczne (prompty) dla konwersacyjnych modeli AI	Załącza odpowiednie pliki, schematy baz danych oraz dokumentację do zapytania w celu eliminacji halucynacji modelu.	Test teoretyczny
	Weryfikuje i iteracyjnie doprecyzowuje odpowiedzi czatu w celu uzyskania kodu zgodnego ze standardami projektu.	Test teoretyczny
Nadzoruje pracę autonomicznego agenta kodującego	Formułuje wieloetapowe zadanie programistyczne w sposób umożliwiający samodzielne wykonanie przez agenta.	Test teoretyczny
	Monitoruje kroki decyzyjne, modyfikacje plików oraz polecenia terminala wykonywane przez narzędzie. Koryguje błędne pętle wykonawcze agenta i modyfikuje plan działania w trakcie realizacji zadania.	Test teoretyczny Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Prowadzi diagnostykę i refaktoryzację oprogramowania przy wsparciu modeli AI	Analizuje logi systemowe i stosy błędów (stack traces) z wykorzystaniem narzędzi generatywnych w celu lokalizacji usterki.	Test teoretyczny
	Przeprowadza refaktoryzację wskazanego modułu kodu pod kątem czytelności i wydajności zgodnie z sugestiami AI.	Test teoretyczny
	Tworzy automatyczne testy jednostkowe i integracyjne pokrywające zmodyfikowaną logikę biznesową.	Test teoretyczny
Weryfikuje poprawność i bezpieczeństwo kodu generowanego maszynowo przed wdrożeniem	Przeprowadza inspekcję bezpieczeństwa wygenerowanych fragmentów pod kątem wstrzykiwania kodu (injection) i wycieków pamięci.	Test teoretyczny
	Odrzuca lub modyfikuje fragmenty kodu niespełniające standardów jakościowych i architektonicznych organizacji.	Test teoretyczny
	Uzasadnia decyzję o manualnej implementacji krytycznych modułów systemu zamiast generowania ich przez AI.	Test teoretyczny
Stosuje procedury ochrony danych i standardy zespołowe w środowisku zintegrowanym z AI	Anonimizuje dane wrażliwe, klucze API oraz dane klientów przed przekazaniem ich do kontekstu modeli AI.	Test teoretyczny
	Rekomenduje zespołowi sprawdzone konfiguracje i szablony promptów optymalizujące pracę grupową.	Test teoretyczny
	Przestrzega polityki wewnętrznej firmy w zakresie dozwolonych narzędzi i poziomów dostępu do kodu źródłowego.	Test teoretyczny

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 rozdzielenie procesów kształcenia i szkolenia od walidacji?

TAK

Program

1. **Mapa narzędzi AI:** Klasyfikacja narzędzi (asystent, czat, agent), zarządzanie kontekstem, koszty, bezpieczeństwo danych i kodu, demonstracja porównawcza na jednym zadaniu.
2. **Asystent w edytorze – GitHub Copilot:** Podpowiedzi kodu, czat w IDE, edycje wieloplikowe w trybie agenta, pliki reguł repozytorium, ćwiczenia praktyczne.
3. **Czat programistyczny – ChatGPT i Claude:** Strukturyzacja promptów technicznych, projektowanie architektury, analiza kodu zastanego (legacy), instrukcje niestandardowe (Custom Instructions/Projects).
4. **Agent kodujący – Claude Code:** Praca z agentem w terminalu, zarządzanie uprawnieniami, plik instrukcji CLAUDE.md, kontrola zmian w systemie Git, ćwiczenia w zadaniu wieloetapowym.
5. **Debugowanie, testy i refaktoryzacja:** Analiza logów i błędów, generowanie testów jednostkowych, refaktoryzacja i optymalizacja modułów, przegląd kodu przed commitem.
6. **Warsztat projektowy i podsumowanie:** Generowanie dokumentacji, standardy zespołowe, praca własna uczestnika z wybranym narzędziem pod okiem trenera, sesja Q&A oraz walidacja wiedzy.

Warunki organizacyjne dla przeprowadzenia usługi:

- **Grupa:** Maksymalnie 25 osób.
- **Samodzielne stanowisko komputerowe:** Każdy uczestnik pracuje na indywidualnym stanowisku (laptop/PC: min. 16 GB RAM, procesor 4-rdzeniowy, OS: Win/macOS/Linux) z uprawnieniami administratora, zainstalowanym środowiskiem IDE, narzędziem Git oraz łączem internetowym min. 50 Mbps.
- **Dostęp i narzędzia:** Zapewniony dostęp do licencji lub bezpłatnych kont narzędzi: GitHub Copilot, Claude / Claude Code, ChatGPT.

Sposób organizacji walidacji:

- **Forma:** Samodzielny, elektroniczny test teoretyczny (quiz online) przeprowadzany w Module 6.
- **Struktura testu:** 10 pytań zamkniętych (jednokrotnego i wielokrotnego wyboru) sprawdzających wiedzę teoretyczną, procedury operacyjne i zagadnienia bezpieczeństwa.
- **Kryterium zaliczenia:** Uzyskanie minimum 60% poprawnych odpowiedzi (12/20 punktów). Wynik generowany jest automatycznie po zakończeniu testu.

Harmonogram

Liczba pozycji harmonogramu: 16

Przedmiot / temat	Typ aktywności	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 16 B1.1 Asystent w edytorze, czat AI i agent kodujący, modele i limity - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	08:00	09:00	01:00
2 z 16 B1.2 Bezpieczeństwo danych, jedno zadanie w trzech narzędziach - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	09:00	10:00	01:00
3 z 16 B2.1 GitHub Copilot: podpowiedzi, czat i kontekst - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	10:00	11:00	01:00
4 z 16 B2.2 GitHub Copilot: tryb agenta, plik instrukcji repozytorium - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	11:00	12:00	01:00
5 z 16 -	Przerwa	-	21-11-2026	12:00	13:00	01:00
6 z 16 B3.1 ChatGPT i Claude: polecenia dla programisty, projektowanie rozwiązań - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	13:00	14:00	01:00

Przedmiot / temat	Typ aktywności	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
7 z 16 B3.2 Wyjaśnianie kodu i dokumentacji, projekty i instrukcje w czacie - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	14:00	15:00	01:00
8 z 16 B4.1 Claude Code: plan, zgody, zmiany w wielu plikach - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	21-11-2026	15:00	16:00	01:00
9 z 16 B4.2 CLAUDE.md, cofanie zmian, wybór agenta lub asystenta - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	22-11-2026	08:00	09:00	01:00
10 z 16 B5.1 Debugowanie z AI, testy jednostkowe i ich weryfikacja - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	22-11-2026	09:00	10:00	01:00
11 z 16 B5.2 Refaktoryzacja i przegląd kodu z AI przed commitem - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	22-11-2026	10:00	11:00	01:00
12 z 16 B6.1 Dokumentacja z AI, zasady pracy z AI w zespole - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	22-11-2026	11:00	12:00	01:00
13 z 16 -	Przerwa	-	22-11-2026	12:00	13:00	01:00

Przedmiot / temat	Typ aktywności	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
14 z 16 B6.2 Warsztat na własnym projekcie, konsultacje indywidualne - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	22-11-2026	13:00	14:00	01:00
15 z 16 B6.3 Warsztat: dokończenie zadania, omówienie rozwiązań, Q&A - współdzielenie ekranu	Zajęcia	PAWEŁ HALLADIN	22-11-2026	14:00	15:00	01:00
16 z 16 -	Walidacja	-	22-11-2026	15:00	16:00	01:00

Podsumowanie

Rodzaj godzin	Liczba godzin
Suma godzin zegarowych usługi	16:00
w tym suma godzin zajęć	13:00
w tym suma godzin walidacji	01:00
w tym suma przerw	02:00
Suma godzin dydaktycznych bez przerw	18:30

Cennik

Jeżeli korzystasz z dofinansowania i usługa stanowi usługę kształcenia zawodowego lub przekwalifikowania zawodowego wraz z usługą lub dostawą towarów ściśle związaną z usługami kształcenia zawodowego lub przekwalifikowania zawodowego to możesz mieć możliwość skorzystania z zwolnienia z podatku VAT na podstawie art. 43 ust. 1 pkt 29 lit. c ustawy z dnia 11 marca 2004 r. o podatku od towarów i usług, jeśli usługa w całości jest finansowana ze środków publicznych lub § 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, gdy usługa jest finansowana w co najmniej 70% ze środków publicznych.

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	1 845,00 PLN
Koszt przypadający na 1 uczestnika netto	1 500,00 PLN
Koszt osobogodziny brutto	115,31 PLN
Koszt osobogodziny netto	93,75 PLN

Liczba godzin usługi

Rodzaj godzin	Liczba godzin
Liczba godzin zegarowych usługi	16:00

Prowadzący

Liczba prowadzących: 1



1 z 1

PAWEŁ HALLADIN

Paweł Halladin – programista i trener IT z ponad 20-letnim doświadczeniem zawodowym. Od 2004 roku projektuje i tworzy aplikacje desktopowe, webowe, mobilne i embedded dla klientów indywidualnych oraz firm. Od 5 lat zawodowo zajmuje się również przekazywaniem wiedzy jako korepetytor i trener IT, prowadząc zajęcia indywidualne oraz szkolenia grupowe dla osób dorosłych, w tym usługi realizowane w ramach Bazy Usług Rozwojowych.

Specjalizuje się w C#/.NET, Java/Spring, C/C++, Python, JavaScript/TypeScript, bazach danych SQL oraz wykorzystaniu AI w pracy programisty. Prowadzi również szkolenia z zakresu aplikacji webowych, embedded/IoT oraz przygotowania do egzaminów zawodowych INF.03 i INF.04.

W latach 2025–2026 zrealizował 195 godzin szkoleń grupowych, w tym 55 godzin szkolenia „Java i Spring w praktyce: projektowanie, testowanie i wdrażanie aplikacji” w ramach BUR oraz 140 godzin szkoleń z C++, .NET i programowania obiektowego. Od 2021 roku prowadzi także indywidualne zajęcia z programowania dla uczniów i studentów.

W pracy szkoleniowej stawia na praktykę – uczestnicy od początku zajęć tworzą działający kod i rozwijają projekty. Złożone zagadnienia wyjaśnia krok po kroku na konkretnych przykładach, dostosowując tempo do poziomu grupy. Posiada swobodę pracy z anglojęzyczną dokumentacją techniczną.

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Materiały:

- projekt ćwiczeniowy z zadaniami do każdego bloku
- przykładowe instrukcje projektu dla Copilota i Claude Code
- zestawienie narzędzi AI dla programistów: zastosowania, koszty, ograniczenia

Warunki uczestnictwa

- komputer stacjonarny lub laptop (Windows 10/11, macOS lub Linux) z działającą kamerą i mikrofonem
- stabilne łącze internetowe o parametrach minimum 10 Mbps download / 5 Mbps upload
- zainstalowany git i edytor kodu (np. Visual Studio Code)
- konto GitHub (darmowe) – dostęp do projektu ćwiczeniowego
- własny dostęp do GitHub Copilot: plan Pro lub wyższy albo licencja firmowa Business / Enterprise (zapewnia uczestnik)
- własny dostęp do ChatGPT: plan Plus lub wyższy (zapewnia uczestnik)
- własny dostęp do Claude i Claude Code: subskrypcja Claude Pro lub wyższa (zapewnia uczestnik)

Warunki techniczne

Każdy uczestnik pracuje na indywidualnym stanowisku (laptop/PC: min. 16 GB RAM, procesor 4-rdzeniowy, OS: Win/macOS/Linux) z uprawnieniami administratora, zainstalowanym środowiskiem IDE, narzędziem Git oraz łączem internetowym min. 50 Mbps.

Kontakt



KRZYSZTOF LISZKA

E-mail liszka.k@gmail.com

Telefon (+48) 509 308 253