



Architektura oprogramowania

Numer usługi 2026/02/08/202247/3315621

4 674,00 PLN brutto

3 800,00 PLN netto

133,54 PLN brutto/h

108,57 PLN netto/h

118,13 PLN cena rynkowa ⓘ

JSYSTEMS SPÓŁKA
Z OGRANICZONĄ
ODPOWIEDZIALNOŚĆ
CIĄ

Brak ocen dla tego dostawcy

📍 zdalna w czasie rzeczywistym

🏠 Usługa szkoleniowa

🕒 35 h

📅 28.09.2026 do 02.10.2026

Informacje podstawowe

Kategoria

Informatyka i telekomunikacja / Programowanie

Grupa docelowa usługi

Szkolenie jest odpowiednie zarówno dla uczestników, którzy mają już doświadczenie w programowaniu, jak i dla tych, którzy chcą usystematyzować i pogłębić swoją wiedzę z obszaru architektury oprogramowania.

Minimalna liczba uczestników

6

Maksymalna liczba uczestników

12

Data zakończenia rekrutacji

24-09-2026

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

Po ukończeniu tego szkolenia uczestnik będzie potrafił:

definiować wymagania pozafunkcjonalne,
wykorzystywać wzorce architektoniczne,
skuteczniej zarządzać ryzykiem w projektach IT,

lepiej optymalizować systemy pod kątem skalowalności i wydajności, sposoby integracji aplikacji, komunikacji i bazach danych, świadome podejście do nowoczesnych technologii, takich jak Microservices, DevOps, CI/CD.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Po ukończeniu tego szkolenia uczestnik będzie potrafił: definiować wymagania pozafunkcjonalne, wykorzystywać wzorce architektoniczne, skuteczniej zarządzać ryzykiem w projektach IT, lepiej optymalizować systemy pod kątem skalowalności i wydajności, sposoby integracji aplikacji, komunikacji i bazach danych, świadome podejście do nowoczesnych technologii, takich jak Microservices, DevOps, CI/CD.	Każde omawiane zagadnienie jest utrwalane rzędem ćwiczeń o wzrastającym poziomie trudności. Podczas szkolenia uczestnicy wykonują ponad 50 warsztatów. W szkoleniu duży nacisk jest kładziony na dobre zrozumienie zasad działania wykorzystywanych technik, a nie ich odtwórcze stosowanie.	Obserwacja w warunkach rzeczywistych

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. Podstawy architektury oprogramowania

o Czym jest architektura i jakie są jej cele

- o Kim jest architekt i jaką pełni rolę
- o Proces i dokumentacja architektoniczna
- o Zarządzanie ryzykiem

2. Cechy jakościowe architektury

- o Parametry systemowe - opis i problemy
- o Poprawne definiowanie wymagań pozafunkcyjnych
- o Charakterystyka parametrów systemowych
 - ☒ Availability, Reliability, Scalability
 - ☒ Performance, Usability, Security
 - ☒ Portability, Reusability, Extensibility
 - ☒ Flexibility, Testability, Realizability
 - ☒ Manageability, Maintainability, Serviceability
- o Wpływ wymiarów systemu na parametry

3. Wzorce architektoniczne

- o Definicja też ważna, czyli jak uniknąć pułapek
- o SDP - Stable Dependency Principle
- o Wsparcie klienta cienkiego: MVC i SPA
- o Podział odpowiedzialności
 - ☒ Desktop, Client-Server
- o 3-Tier, 4-Tier, Layers pattern
- o Wzorce infrastruktury
 - ☒ Skalowanie poziome i pionowe
 - ☒ Modularyzacja, Load Balancing
 - ☒ Redundancja Ścieżek, Cloud
 - ☒ Klastry HA i Failover
- o Wzorce Enterprise Application Integration
 - ☒ MOM (message broker), SOA, ESB
- o Microservices
- o Wzmianka o szablonach: Microservices Patterns, POSA, PEAA, DDD, EIP

4. Więcej o Microservices

- o To SOA czy nie SOA?
- o Monolit jako przeciwległa skrajność
- o DevOps i CI/CD
- o Są zalety, są problemy
- o On Premise(s), IaaS, CaaS, PaaS, FaaS, SaaS
- o Partycjonowanie w skalowaniu 3D

o Rady, przeciwwskazania i przeciwne zdania

o Polyglot Persistence

o CQRS i Event Sourcing

o ACID kontra BASE

o Kiedy migrować do microservices

5. Architektura warstwy klienta i prezentacji

o Gdzie przechowywać sesję

o Rodzaje aplikacji klienckich

o Technologie klienta grubego: biblioteki, RCP

o Technologie klienta cienkiego

☒ Strony statyczne i dynamiczne

☒ Frameworki MVC

☒ Frameworki SPA: Angular, React, Vue

☒ Komunikacja zaawansowana: AJAX i WebSocket

6. Architektura warstwy biznesowej

o Rozpraszać czy nie

o Wydajność komunikacji zdalnej a lokalnej

o Protokoły komunikacyjne

☒ CORBA jako przykład długu technologicznego

☒ SOAP, REST i GraphQL

☒ Sockets czyli protokół własny

☒ RMI jako przykład rozwiązania homogenicznego

☒ gRPC i inne rozwiązania rynkowe

o Kryteria wyboru serwera aplikacyjnego

7. Architektura warstwy integracji i zasobów

o Technologie utrwalania danych

☒ Bazy relacyjne

☒ Bazy hierarchiczne

☒ Różne modele danych w NoSQL

☒ Big data i Data Lake

o Komunikacja asynchroniczna

o Bezpośrednia, ESB, MOM

☒ Wzorce EIP i ich wsparcie

☒ API kontra protokół, czyli JMS a AMQP

☒ Przykładowe rozwiązania: RabbitMQ, Redis, Kafka

☒ Różnice między implementacjami message brokera

- o Legacy w teorii i praktyce
- o Ostatnia deska ratunku: Screen Scraping
- o Scaffolding, czyli jak to wszystko zwi zać

8. Dokumentowanie architektury w UML

- o Opisowy diagram komponent w
- ☒ Komponenty proste i zło zone
- ☒ Interfejs wymagany i dostarczany
- ☒ Złączenie i zale ność
- ☒ Porty
- o Diagram wdrożenia
- ☒ Węzły i ich stereotypy
- ☒ Communication path - opis, liczno ć, kierunkowo ć

9. Wgląd w zaawansowane modelowanie architektury

- o Dodatkowe wymiary diagramu wdrożenia
- o Instancyjny
- o Niskopoziomowy
- o Szablon architektoniczny
- o Wdrażanie artefakt w
- o Diagram pakiet w
- ☒ Pakiet
- ☒ Access a import

10. Uwzględnienie architektury w projekcie

- o Warstwy i komponenty a...
- o Model projektowy w UML
- o Fizyczna realizacja projektu
- o Projekt a ograniczenia architektury

11. Weryfikacja i ocena architektury

- o Korzyści weryfikacji
- o Zesp ł weryfikujący
- o Techniki, proces i raport z weryfikacji

Harmonogram

Liczba przedmiot w/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	4 674,00 PLN
Koszt przypadający na 1 uczestnika netto	3 800,00 PLN
Koszt osobogodziny brutto	133,54 PLN
Koszt osobogodziny netto	108,57 PLN

Prowadzący

Liczba prowadzących: 1



1 z 1

Michał Podbielski

Programista Java od 2004 roku, doświadczony w wielkich międzynarodowych projektach. Architekt oprogramowania i konsultant w zakresie architektury oprogramowania, UML i BPMN rozwiązujący problemy klientów w złożonych i zróżnicowanych projektach. Przeprowadził w sumie ponad 390 szkoleń ze świetnymi wynikami w ankietach. Prowadzi szkolenia z języka Java, architektury oprogramowania, UML i BPMN.

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Uczestnicy otrzymają komplet materiałów PDF

Warunki uczestnictwa

Podstawowa znajomość programowania, umiejętność korzystania z komputera.

Informacje dodatkowe

Szkolenie kończy się wydaniem certyfikatu imiennego

Warunki techniczne

Dostęp do internetu, laptop/komputer, wideoaplikacja ZOOM lub dostęp do wersji w przeglądarce, możliwość podłączenia się do hostów przygotowanych na pulpicie zdalnym - połączenie przez RDP.

Kontakt



Biuro Obsługi Klienta

E-mail biuro@jsystems.pl

Telefon (+48) 534 506 503