



## Spring Framework - szkolenie kompleksowe

Numer usługi 2026/02/12/202247/3330831

3 690,00 PLN brutto

3 000,00 PLN netto

105,43 PLN brutto/h

85,71 PLN netto/h

157,50 PLN cena rynkowa ⓘ

JSYSTEMS SPÓŁKA  
Z OGRANICZONĄ  
ODPOWIEDZIALNOŚ  
CIĄ

Brak ocen dla tego dostawcy

- 📄 Usługa szkoleniowa
- 📺 zdalna w czasie rzeczywistym
- 🕒 35:00 h
- 📅 23.11.2026 do 27.11.2026

## Informacje podstawowe

|  |   |
|--|---|
| <b>Kategoria</b>                       | Informatyka i telekomunikacja / Programowanie   |
| <b>Grupa docelowa usługi</b>           | Szkolenie jest przeznaczone dla osób, które znają podstawy programowania w języku Java i chcą tworzyć aplikacje webowe/korporacyjne z wykorzystaniem Spring Framework, w szczególności jeśli planują pracę nad projektami backendowymi lub chcą rozwinąć kompetencje w nowoczesnych technologiach Java. |
| <b>Minimalna liczba uczestników</b>    | 6   |
| <b>Maksymalna liczba uczestników</b>   | 10  |
| <b>Data zakończenia rekrutacji</b>     | 19-11-2026  |
| <b>Forma prowadzenia usługi</b>        | zdalna w czasie rzeczywistym  |
| <b>Liczba godzin usługi</b>            | 35  |
| <b>Podstawa uzyskania wpisu do BUR</b> | Certyfikat systemu zarządzania jakością wg. ISO 9001:2015 (PN-EN ISO 9001:2015) - w zakresie usług szkoleniowych  |

## Cel

### Cel edukacyjny

Nabycie przez uczestników kompleksowych umiejętności tworzenia aplikacji enterprise przy użyciu Spring Framework i Spring Boot, umożliwiających samodzielne projektowanie i budowanie skalowalnych aplikacji serwerowych z warstwą REST API, dostępem do danych i bezpieczeństwem.

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

| Efekty uczenia się  | Kryteria weryfikacji   | Metoda walidacji                                      |
|---|--|---|
| Tworzy aplikacje Spring Boot z konfiguracją auto-configuration.     | Uczestnik inicjalizuje projekt Spring Boot, konfiguruje zależności Maven/Gradle i uruchamia aplikację webową z co najmniej jednym kontrolerem REST, weryfikując działanie przez Postman. | Test teoretyczny z wynikiem generowanym automatycznie |
| Implementuje warstwę dostępu do danych przy użyciu Spring Data JPA. | Uczestnik tworzy encję JPA, repozytorium Spring Data i implementuje operacje CRUD oraz co najmniej jedno zapytanie custom (JPQL lub derived query), testując na H2 lub PostgreSQL.       | Test teoretyczny z wynikiem generowanym automatycznie |
| Zabezpiecza aplikację przy użyciu Spring Security.                  | Uczestnik konfiguruje uwierzytelnianie JWT w Spring Security, chroni wybrane endpointy dla ról użytkowników i weryfikuje zasady dostępu w testach integracyjnych.                        | Test teoretyczny z wynikiem generowanym automatycznie |
| Pisze testy jednostkowe i integracyjne dla aplikacji Spring         | Uczestnik tworzy testy jednostkowe serwisu przy użyciu JUnit 5 + Mockito oraz test integracyjny endpointu REST z użyciem @SpringBootTest i MockMvc, wszystkie przechodzące pomyślnie.    | 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 rozdzielenie procesów kształcenia i szkolenia od walidacji?

TAK

# Program

## 1. Spring container

o Wzorzec odwróconego sterowania i wstrzykiwanie zależności:

Kluczowa koncepcja zarządzania zależnościami między komponentami.

o Kontekst aplikacyjny i kontener Spring'a:

Środowisko zarządzające cyklem życia i konfiguracją obiektów.

o Konfiguracja XML oraz adnotacyjna:

Różne podejścia do konfigurowania komponentów i ich zależności.

o Cykl życia komponentów:

Proces tworzenia, inicjalizacji i niszczenia komponentów Spring.

o Automatyczne wiązania:

Mechanizmy automatycznego kojarzenia zależności.

o Spring Expression Language:

Język wyrażeń do dynamicznego ustawiania wartości w komponentach.

## 2. Programowanie Aspektowe - Spring AOP

o Założenia AOP:

Wprowadzenie do programowania aspektowego i jego głównych konceptów.

o Obsługa programowania aspektowego w Springu:

Integracja mechanizmów AOP w ramach Spring.

o Punkty złączenia, punkty przecięcia:

Miejsca w kodzie, w których można wstrzykiwać logikę aspektów.

o Porady @Before, @After i @Around:

Różne typy porad definiujących, kiedy wywoływać aspekty.

o Procesowanie adnotacji przy użyciu AOP:

Wykorzystywanie aspektów jako procesor adnotowanych metod

## 3. Spring MVC

o Wzorzec MVC:

Organizacja aplikacji w logiczne warstwy Model-Widok-Kontroler.

o Inicjalizacja serwleta:

Konfiguracja serwleta jako punktu wejścia aplikacji.

o Kontrolery i mapowanie wywołań:

Definiowanie logiki obsługującej żądania HTTP.

o Widok:

Generowanie dynamicznego HTML na podstawie danych z kontrolera.

o Przekazywanie modelu do warstwy widoku:

Mechanizm wymiany danych między warstwami.

o Parametry żądania:

Obsługa parametrów wejściowych z zapytań HTTP.

o Wykorzystanie plików JSP i tagów JSTL we współpracy ze Springiem:

Tworzenie dynamicznych stron.

o Wielojęzyczność w Springu:

Mechanizmy wspierające różne wersje językowe aplikacji.

o Interceptory:

Przechwytywanie i modyfikowanie żądań na poziomie kontrolera.

o Formularze:

Obsługa danych przesyłanych z formularzy HTML.

o Walidacja formularzy:

Weryfikacja poprawności danych wejściowych.

#### **4. Usługi REST**

o Czym są usługi REST?:

Zasady projektowania usług webowych bazujących na architekturze REST.

o Obsługa REST w Springu:

Implementacja API REST w klasach kontrolerów Spring.

o Obsługa adresów URL typu RESTful:

Definiowanie adresów URL odpowiadających zasobom.

o Reprezentacja zasobów:

Formatowanie danych w odpowiedzi na żądania (np. JSON, XML).

o Tworzenie kontrolerów korzystających z zasobów:

Projektowanie API opartych na REST.

o Tworzenie klientów REST:

Konsumpcja zewnętrznych API w aplikacji Spring.

o Walidacja danych:

Weryfikacja danych przesyłanych przez klienta.

#### **5. Spring JDBC, Integracja z Hibernate i JPA**

o JDBC w Springu:

Obsługa bezpośredniego dostępu do bazy danych.

o Szablony dostępu do danych:

Upraszczenie operacji na bazach danych.

o Klasy bazowe repozytoriów DAO:

Struktura umożliwiająca łatwe operacje CRUD.

o Wyjątki związane z dostępem do danych:

Obsługa błędów w komunikacji z bazą danych.

o Źródła danych:

Zarządzanie połączeniami do bazy danych.

o Wykorzystanie puli połączeń:

Optymalizacja pracy z bazą danych.

o Wydajność zapytań i jej poprawa:

Techniki optymalizacji zapytań.

o Integracja JPA ze Springiem ze wsparciem Hibernate:

Mapowanie obiektowo-relacyjne (ORM).

o Deklaratywne repozytoria z użyciem Spring Data

Automatyzacja implementacji klas repozytoryjnych

## **6. Spring Security**

o Architektura Spring Security:

Warstwa ochrony w aplikacjach Spring.

o Zabezpieczania HTTP:

Kontrola dostępu do zasobów na poziomie HTTP.

o Zabezpieczanie elementów na poziomie widoku:

Ukrywanie funkcji interfejsu dla nieautoryzowanych użytkowników.

o Zabezpieczanie metod:

Kontrola dostępu na poziomie logiki aplikacji.

o Repozytorium użytkowników:

Zarządzanie danymi użytkowników.

o Uwierzytelnianie:

Proces weryfikacji tożsamości użytkowników.

## **7. Spring Boot**

o Konwencja ponad konfigurację:

Automatyzacja konfiguracji aplikacji.

o Uruchamianie aplikacji Spring Boot:

Tworzenie gotowych do działania aplikacji w jednym kroku.

o Zależności typu Starter:

Predefiniowane zależności upraszczające konfigurację.

o Konsolidacja zależności:

Zarządzanie zależnościami w projekcie.

o Plik konfiguracyjny:

Centralne miejsce zarządzania ustawieniami aplikacji.

o Monitoring stanu aplikacji przy użyciu Actuator'a:

# Harmonogram

Liczba pozycji harmonogramu: 16

| Przedmiot / temat  | Prowadzący      | Data realizacji zajęć | Godzina rozpoczęcia | Godzina zakończenia | Liczba godzin |
|--|-----------------|-----------------------|---------------------|---------------------|---------------|
| <b>1 z 16</b> Spring container                           | Marcin Dzierżak | 23-11-2026            | 09:00               | 12:30               | 03:30         |
| <b>2 z 16</b> Przerwa Obiadowa                           | Marcin Dzierżak | 23-11-2026            | 12:30               | 13:00               | 00:30         |
| <b>3 z 16</b> Spring Expression Language                 | Marcin Dzierżak | 23-11-2026            | 13:00               | 16:00               | 03:00         |
| <b>4 z 16</b> Programowanie Aspektowe - Spring AOP       | Marcin Dzierżak | 24-11-2026            | 09:00               | 12:30               | 03:30         |
| <b>5 z 16</b> Przerwa Obiadowa                           | Marcin Dzierżak | 24-11-2026            | 12:30               | 13:00               | 00:30         |
| <b>6 z 16</b> Procesowanie adnotacji przy użyciu AOP     | Marcin Dzierżak | 24-11-2026            | 13:00               | 16:00               | 03:00         |
| <b>7 z 16</b> Spring MVC                                 | Marcin Dzierżak | 25-11-2026            | 09:00               | 12:30               | 03:30         |
| <b>8 z 16</b> Przerwa Obiadowa                           | Marcin Dzierżak | 25-11-2026            | 12:30               | 13:00               | 00:30         |
| <b>9 z 16</b> Usługi REST                                | Marcin Dzierżak | 25-11-2026            | 13:00               | 16:00               | 03:00         |
| <b>10 z 16</b> Spring JDBC, Integracja z Hibernate i JPA | Marcin Dzierżak | 26-11-2026            | 09:00               | 12:30               | 03:30         |
| <b>11 z 16</b> Przerwa Obiadowa                          | Marcin Dzierżak | 26-11-2026            | 12:30               | 13:00               | 00:30         |
| <b>12 z 16</b> Spring Security                           | Marcin Dzierżak | 26-11-2026            | 13:00               | 16:00               | 03:00         |

| Przedmiot / temat                                 | Prowadzący      | Data realizacji zajęć | Godzina rozpoczęcia | Godzina zakończenia | Liczba godzin |
|---|-----------------|-----------------------|---------------------|---------------------|---------------|
| <b>13 z 16</b> Spring Boot                        | Marcin Dzierżak | 27-11-2026            | 09:00               | 12:30               | 03:30         |
| <b>14 z 16</b> Przerwa Obiadowa                   | Marcin Dzierżak | 27-11-2026            | 12:30               | 13:00               | 00:30         |
| <b>15 z 16</b> Zajęcia szkoleniowe – podsumowanie | Marcin Dzierżak | 27-11-2026            | 13:00               | 15:30               | 02:30         |
| <b>16 z 16</b> Walidacja                          | Marcin Dzierżak | 27-11-2026            | 15:30               | 16:00               | 00:30         |

## Cennik

### Cennik

| Rodzaj ceny                               | Cena         |
|---|--------------|
| Koszt przypadający na 1 uczestnika brutto | 3 690,00 PLN |
| Koszt przypadający na 1 uczestnika netto  | 3 000,00 PLN |
| Koszt osobogodziny brutto                 | 105,43 PLN   |
| Koszt osobogodziny netto                  | 85,71 PLN    |

## Prowadzący

Liczba prowadzących: 1



**1 z 1**

### Marcin Dzierżak

Trener posiada wieloletnie doświadczenie w tworzeniu aplikacji enterprise z użyciem Spring Framework, przy czym kluczowe kwalifikacje w zakresie Spring Boot, Spring MVC, Spring Security i architektury mikroserwisowej zostały zdobyte i są czynnie wykorzystywane w okresie ostatnich 5 lat (od 2021 roku do chwili obecnej). Potwierdzają to zrealizowane projekty developerskie dla sektora finansowego i korporacyjnego oraz szkolenia z Spring Framework 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 języka Java, 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@jsystems.pl](mailto:biuro@jsystems.pl)

**Telefon** (+48) 534 506 503