

## Program szkolenia:

# Mikroserwisy w Microsoft Azure

### Informacje:

<b>Nazwa:</b>	<b>Mikroserwisy w Microsoft Azure</b>
<b>Kod:</b>	<b>azure-ms</b>
<b>Kategoria:</b>	Azure
<b>Odbiorcy:</b>	developerzy
<b>Czas trwania:</b>	3 dni
<b>Forma:</b>	20% wykłady / 80% warsztaty

---

Szkolenie jest praktycznym warsztatem pokazującym jak wykorzystać usługi w Microsoft Azure do stworzenie systemu opartego na architekturze mikroserwisów.

Podczas tego szkolenia zaimplementujemy kilka systemów z architekturą mikosewisów hostowanych w różnych usługach. Poznamy zalety, wady i ograniczenia każdego sposobu oraz jak łączyć je w jednym systemie.

Uczestnicy poznają usługi i wzorce niezbędne w architekturze mikroserwisowej, takie jak: monitoring, automatyczne skalowanie (wertykalne i horyzontalne), kolejki, wsparcie dla działań DevOps są dostępne jako usługi.

### Zalety szkolenia:

- Świadome stosowanie wzorców i technologii z uwzględnieniem kontekstu biznesowego
- Elementy DDD: Bounded Context
- Kompleksowe podejście do architektury i jej składowych

## Szczegółowy program:

### 1. Wstęp

1.1. Co to jest architektura mikroserwisów?

1.2. Jaki problem rozwiązują mikroserwisy?

1.3. Czy mikroserwis to zawsze dobra decyzja?

1.4. Wzorce i antywzorce.

1.4.1. Database per Service pattern describes how each service has its own database in order to ensure loose coupling.

1.4.2. API Gateway

1.4.3. Service Discovery

1.4.4. Messaging vs. Remote Procedure Invocation

1.4.5. Single Service vs. Multiple Services per Host

1.4.6. Microservice chassis pattern and Externalized configuration

1.4.7. Service Component Test

1.4.8. Service Integration Contract Test

1.4.9. Circuit Breaker

1.4.10. Access Token

1.4.11. Log aggregation

1.4.12. Application metrics

1.4.13. Audit logging

1.4.14. Distributed tracing

1.4.15. Exception tracking

1.4.16. Health check API

1.4.17. Log deployments and changes

1.4.18. Server-side page fragment composition

1.4.19. Client-side UI composition

1.5. Wyznaczanie granic serwisu

1.5.1. Business Capabilities

1.5.2. Bounded Context

## 2. App Services

2.1. Architektura i zastosowanie

2.2. Troubleshooting

2.3. Komunikacja synchroniczna (HTTP)

2.4. Kudu i debugowanie

2.5. Deployment slots

2.6. Monitoring

## 3. Azure Functions

3.1. Architektura serverless

3.2. HTTP trigger functions

3.3. Message trigger functions

3.4. Debugowanie

3.5. Deployment

3.6. Durable functions

## 4. Service Fabric

4.1. Architektura

4.2. Zastosowanie

4.3. Containers

4.4. Reliable Services

4.5. Reliable Actors

4.6. Guest executables

4.7. Klastry

4.8. Lokalny development

4.9. Skalowanie

4.10. Monitoring

## 5. AKS (Azure Kubernetes Services)

5.1. Architektura

5.2. Zastosowanie

5.3. Klastry

5.4. Dostęp i zarządzanie uprawnieniami

5.5. Sieć

5.6. Skalowanie

5.7. Monitoring