

Tytuł szkolenia: RabbitMQ – systemy kolejkowe

Kod szkolenia: RabbitMQ

Wprowadzenie

Czas trwania: 1 do 2 dni, w zależności od aktualnego doświadczenia uczestników z RabbitMQ. Szkolenie może zostać przeprowadzone zarówno w języku polskim jak i angielskim oraz może być dostosowane do indywidualnych potrzeb i wymagań grupy.

Adresaci szkolenia

Cel szkolenia

- Umiejętność uruchomienia oraz pracy z RabbitMQ.
- Umiejętność dopasowania parametrów kolejek oraz message'y do wymagań biznesowych.
- Umiejętność budowania niezawodnych systemów (reliability).
- Umiejętność konfiguracji klastra RabbitMQ.
- Umiejętność praktycznego użycia RabbitMQ i integracji z istniejącym oprogramowaniem (unikanie tzw. golden hammer).
- Umiejętność pracy z systemami kolejkowymi.
- Testowanie wydajności rozproszonego systemu i odnajdywania wąskich gardeł.
- Umiejętność wyboru gotowych na rynku narzędzi i frameworków celem optymalizacji pracy developera – użycie technologii, które do tej pory nie były brane pod uwagę, a z sukcesem mogą zostać wdrożone.

Cena szkolenia różni się w zależności od ilości dni szkolenia. **Skontaktuj się z nami, aby otrzymać dedykowaną ofertę.**

Czas i forma szkolenia

- 14 godzin (2 dni x 7 godzin), w tym wykłady i warsztaty praktyczne.

Plan szkolenia

1. Wprowadzenie

- o Dlaczego świat ekscytuje się BigData i co to jest? – wprowadzenie w świat technologii BigData
- o Własny DC czy Chmura - przegląd plusów i minusów w kontekście Amazon Web Services (AWS)
- o Przegląd dostępnych na rynku systemów kolejkowych (tj.: RabbitMQ, Kafka, AWS Kinesis itp.) – *jakich produktów należy używać i kiedy, jakie są różnice między nimi, jakie są wady i zalety poszczególnych z nich*
- o Open Source vs Closed Source – *dlaczego warto wybierać technologie OpenSource*

2. RabbitMQ – wprowadzenie

- o Protokół AMQP
- o Architektura produktu RabbitMQ
- o Środowisko pracy – *Windows vs Unix, Erlang*
- o Producer, Consumer, Queue, Broker, Exchange, Binding, Topics, Routing Keys – *elementy RabbitMQ, których zrozumienie jest kluczowe*
- o Queues – durability, priorities, exclusive, persistence, TTLs, lazy queues, Dead-Lettered queues, publish-subscribe, acknowledgments, transactions

3. RabbitMQ – standalone instance

- o Uruchomienie standalone instance
- o Warsztaty z przykładowego użycia (Simple Queue, Working Queue, Publish/Subscribe bazujące na dowolnym kluczu, routing key, topics czy headers)
- o RPC
- o Policies

4. RabbitMQ – budowanie klastra

- o Skalowalność – *Vertical scaling vs Horizontal scaling*
- o Konfiguracja nodów w klastrze
- o Fault tolerance – odporność na awarie, HA (mirrored queues), Quorum Queues, Federation, Shovel, Partitions

5. RabbitMQ – WebAdmin

6. RabbitMQ – Monitoring

7. RabbitMQ – Security/Permissions

8. Rozszerzony program warsztatów