

# Checklista CTO przed wybo**rem** lub **modernizacją** Java

- 1 Czy system obsługuje proces krytyczny dla przychodów?
- 2 Czy obecna architektura pozwala skalować produkt?
- 3 Czy mamy problem z kosztem utrzymania?
- 4 Czy system działa na wspieranej wersji Java?
- 5 Czy znamy poziom technical debt?
- 6 Czy mamy automatyczne testy regresji?
- 7 Czy deployment jest powtarzalny?
- 8 Czy zespół ma seniorów Java?
- 9 Czy dokumentacja architektury jest aktualna?
- 10 Czy znamy koszt infrastruktury per usługa?
- 11 Czy system ma observability?
- 12 Czy istnieje plan migracji do nowszej wersji Java?
- 13 Czy architektura wspiera integracje AI?
- 14 Czy ryzyka bezpieczeństwa są monitorowane?
- 15 Czy system można rozwijać bez zależności od jednej osoby?
- 16 Czy vendor lock-in jest kontrolowany?
- 17 Czy decyzja Java vs alternatywa została oceniona biznesowo?
- 18 Czy mamy plan modernizacji bez zatrzymania biznesu?
- 19 Czy koszt ownership jest policzony na 3 lata?
- 20 Czy technologia wspiera roadmapę produktu?