

# CTO Checklist Before **Choosing** or **Modernizing** a Java Platform

- 1 Does the system support a revenue-critical business process?
- 2 Does the current architecture allow the product to scale?
- 3 Do we have an issue with maintenance costs?
- 4 Is the system running on a supported Java version?
- 5 Do we know the level of technical debt?
- 6 Do we have automated regression testing?
- 7 Is the deployment process repeatable and reliable?
- 8 Does the team include senior Java engineers?
- 9 Is the architecture documentation up to date?
- 10 Do we know the infrastructure cost per service?
- 11 Does the system have adequate observability?
- 12 Is there a roadmap for migrating to a newer Java version?
- 13 Does the architecture support AI integrations?
- 14 Are security risks continuously monitored?
- 15 Is the system free from single-person dependency?
- 16 Is vendor lock-in being actively managed?
- 17 Has Java vs. alternatives been assessed?
- 18 Do we have a modernization plan without business disruption?
- 19 Is the 3-year TCO calculated?
- 20 Does the technology support the product roadmap?