

Year-1 Decisions to Get Right

Florinel Chis — Magendoo Interactive

magendoo.ro/guides/why-magento-projects-fail

Year-1 Decisions to Get Right

From the guide: Why Most Magento Projects Fail in Year 2

- [] 1. Integration logic lives outside Magento — middleware, message queues, or dedicated services
 - [] 2. Every custom module has a documented justification: unique business requirement, not a workaround
 - [] 3. Performance budget defined: max DB queries per page, no sync external calls in checkout critical path
 - [] 4. No direct database access from custom code — use Magento repositories and collections
 - [] 5. Around plugins used sparingly; alternatives (after plugins, event observers) preferred
 - [] 6. Cron jobs are not integration orchestration — they're scheduled tasks with narrow scope
 - [] 7. At least two developers understand every integration point in the system
 - [] 8. Architecture decisions documented in a living ADR (Architecture Decision Record) file
 - [] 9. Code review explicitly checks for: query correctness, cache declarations, observer scope
 - [] 10. Upgrade path tested quarterly — run 'composer outdated' and assess the delta
-

Based on 22+ years and 50+ enterprise Magento projects.

Full guide: magendoo.ro/guides/why-magento-projects-fail

Need help? magendoo.ro/contact

© 2026 Magendoo Interactive SRL — magendoo.ro