

Laravel Cloud Migration Readiness Checklist

Architecture

- Is your app monolithic, or do you have microservices?
- Do you store files locally, or can they go to a cloud bucket like S3?
- Are you ready to shift session and cache handling to a shared resource (like Redis)?

Cost & Budget

- Are you okay trading a predictable monthly server bill for a usage-based model?
- Does your app handle big traffic spikes (which might increase costs)?
- Have you compared the potential DevOps savings against any new charges (like bandwidth or storage fees)?

Performance & Monitoring

- How will you track performance metrics and pinpoint bottlenecks (e.g., built-in tools vs. third-party APM)?
- Do you plan to run load or stress tests to see how the app scales under heavy traffic?
- Are you prepared to set up real-time alerts for critical metrics (CPU, memory, response times) to prevent downtime?
- Will you use a centralized logging solution to store and analyze logs from various services?
- Do you need any specific monitoring or alerting integrations (e.g., Slack, email, incident management systems)?

Compliance & Policy

- Are you in a regulated industry (finance, health) that needs HIPAA, SOC 2, or GDPR alignment?
- Does the Laravel Cloud provider meet those requirements or offer specific compliance guarantees?

Security & Data Protection

- Do you handle sensitive data that requires encryption at rest or in transit?
- Who manages SSL certificates—does the platform automate renewals or do you handle them manually?
- How are secrets (API keys, credentials) stored and rotated—environment variables, vault solutions, etc.?
- What's your plan for backup and disaster recovery—does the platform provide automatic snapshots?
- Do you need additional intrusion detection or advanced threat protection beyond standard measures?

Team Readiness

- Is your team comfortable with new deployment process (CI/CD pipelines, environment variables, debugging logs, etc.)?
- Will DevOps folks shift to more strategic tasks, or do they need training?
- How comfortable are non-technical stakeholders with potential downtime during migration?

Timeline & Downtime

- How long can your app be offline during the move?
- Will you do a cutover approach (shut down the old system, migrate data, then turn on the new one) or a more gradual swap?
- Do you have a rollback plan if something goes off the rails after flipping the DNS?

Future Flexibility

- Are you okay with potentially getting "locked in" to a specific platform's tooling?
- Will the process be a hassle if you ever want to move off?
- Have you considered the long-term trade-offs before you commit?