SHANNON LLC ｜ CI/CD

CI/CD構築

サービス資料

手動デプロイから、「押せば本番へ」。テスト・ビルド・デプロイの自動化を、現状の開発スタックに合わせて設計します。

手動デプロイのままでは、リリース頻度の頭打ちと手順の属人化を招きます

こんなお悩み、ありませんか？　1つでも当てはまれば、CI/CD 整備を検討するタイミングです。

属人化した手動デプロイ

本番デプロイが手動で、リリースのたびに「あの人」を呼び出している。手順を知る担当者が不在だと、リリース自体が止まってしまう。

テストがCIで動かない

テストは書いてあるが、CI で実行されておらず、リリース時に通すだけ。コミット時点で検知できず、問題の発見がリリース直前にずれ込む。

ロールバック手順が無い

障害が起きたとき、ロールバックの手順が決まっていない。復旧が担当者の記憶頼みになり、切り戻しに時間がかかってしまう。

環境差異による不具合

staging 環境と本番環境の差異で、リリース後にしか気づかない不具合がある。検証を通っても本番で壊れるため、リリースのたびに緊張が走る。

リリース頻度の頭打ち

「リリース頻度を上げたい」が体制が追いつかない。手動デプロイのまま頻度だけ上げると、作業ミスによる事故が起きやすくなる。

誰も触れないCI/CD設定

CI/CD の設定が古く、誰も触れない状態になっている。作った担当者が離れてしまい、失敗しても原因を調べられない。

これらの悩みは根が同じです。テスト・ビルド・デプロイが「仕組み」ではなく**「人の手順」に依存**していること。次ページで、CI/CD が効くタイミングを整理します。

Copyright © 2026 SHANNON LLC. All rights reserved.

CI/CD が "効く" のは、増員・リリース頻度向上・障害削減に向き合うときです

CI/CD を後回しにできるのは、コードベースが小さく開発者が 1〜2 名のうちだけです。

01

開発者を増やすとき

コミット数が増えると、テスト未実行のままマージするリスクが急増。CI が無いと「人がレビューで全部見る」前提になり、レビュー疲弊を招きます。

- コミットごとにテスト・ビルドを自動実行
- ユニットテスト自動実行・静的解析

02

リリース頻度を上げたいとき

週次 → 日次にしたい、1日に複数回リリースしたい。手動デプロイを残したまま頻度だけ上げると、作業ミスによる事故が起きやすくなります。

- 検証済み成果物の自動デプロイ（CD）
- 承認フロー・ロールバック自動化

03

本番障害を減らしたいとき

「テスト書いてるのに障害が出る」のは、CI で実行されていないかカバレッジが偏っているかのどちらか。CI 整備がスタートラインです。

- カバレッジ計測と閾値の設定
- 失敗時の切り戻し手順の明文化

※ CI＝継続的インテグレーション（コミットごとのテスト・ビルド自動実行）／CD＝継続的デリバリー（検証済み成果物の自動デプロイ）。

GitHub Actions / CircleCI / CodePipeline / GitLab CI から、リポジトリ規模と組織の運用体制に合わせて選定。本番障害の不安なくリリース頻度を上げる体制を構築します。

Copyright © 2026 SHANNON LLC. All rights reserved.

対応範囲①

ツール選定・テスト・デプロイ自動化を、必要な範囲だけ実装

既存リポジトリの状態を踏まえて設計します。使わない機能まで一式で導入することはしません。

ツール選定・設計

リポジトリの場所・組織の運用体制・既存資産に応じて最適なツールを選定します。「GitHub Actions一択」と決め打ちせず、既存資産が活きる構成を提案します。

- GitHub Actions（セルフホスト可）
- CircleCI / GitLab CI
- AWS CodePipeline / CodeBuild
- Bitbucket Pipelines
- セルフホストランナー設計
- 既存資産との両立

テスト自動化

ユニット・E2E・静的解析を CI に組み込みます。既存のテスト資産を尊重しつつ、足りない部分のテストピラミッド設計もあわせて行います。

- ユニットテスト自動実行
- Playwright / Cypress E2E
- 静的解析（ESLint / PHPStan / RuboCop）
- セキュリティスキャン（Snyk / Trivy）
- カバレッジ計測と閾値
- 並列実行・キャッシュ最適化

デプロイ自動化

ビルド成果物を本番 / staging / dev に自動配信します。Blue-Green / Rolling / Canary など、サービス特性に応じたデプロイ戦略を組みます。

- マルチ環境（dev / stg / prod）
- Blue-Green / Rolling デプロイ
- 承認フロー（手動承認ステップ）
- ロールバック自動化
- ECS / Lambda / EC2 / 静的サイト
- WordPress / WP-CLI 連携

※ Blue-Green / Rolling ＝ 無停止リリースのためのデプロイ方式。Canary 配信を含め、サービス特性に応じて選定します。

Copyright © 2026 SHANNON LLC. All rights reserved.

対応範囲②

通知・移行・コスト最適化まで、運用に乗る形に整えます

「作って終わり」にせず、気づける通知・段階的な移行計画・CI 利用料の圧縮まで踏み込みます。

通知・運用

失敗・成功・レビュー依頼を普段使いのチャットに通知し、パイプラインの状態に気づける体制を整えます。過剰通知の絞り込みも行います。

- Slack / Google Chat / Chatwork 通知
- 失敗通知の絞り込み（過剰通知の解消）
- PR レビュー自動アサイン
- リリース連絡の自動化
- エラー監視ツール連携
- 運用手順書の整備

既存 CI からの移行

古い Jenkins や、設定が散らばって誰も触れない CI を、既存資産を活かしつつモダンな構成へ段階的に移行。並行稼働で切り替えリスクを抑えます。

- Jenkins / 古い CI からの移行
- 設定の棚卸し・ドキュメント化
- 並行稼働期間の運用
- 差分検証・互換性確認
- カットオーバー計画
- 切り戻し手順の明文化

コスト・ランナー最適化

実行時間・並列度・キャッシュ・セルフホストランナーの活用で、CI 利用料を圧縮します。月額費用が膨らんできた組織のコスト削減に有効です。

- 実行時間・分単価の現状把握
- キャッシュ戦略の見直し
- 並列実行の最適化
- セルフホストランナー導入
- 不要ジョブの整理
- 月次 CI コストレポート

※ セルフホストランナー＝自社サーバー・社内ネットワーク内で CI ジョブを実行する方式。月額が膨らんできた組織のコスト削減にも有効です。

Copyright © 2026 SHANNON LLC. All rights reserved.

テンプレ導入や付帯CIと違い、診断から運用まで「資産が残る」形で構築します

よくある2つの選択肢と、シャノンの進め方の違いを整理しました。分かれ目は「資産が残るか」です。

自社で対応	クラウドベンダー付帯	SHANNON ｜ 診断 → 設計 → 構築 → 運用
✕テンプレを写して導入したまま、改善されずそれっきりになりがち	✕構成がベンダーの自社製品前提に偏り、他への乗り換えが難しい	✓リポジトリ・テスト資産・デプロイ手順を棚卸しして診断・設計
✕テスト・デプロイ・通知の連携が中途半端に終わる	✕テストの整備までは範囲に入らず、通知設計も別契約になりがち	✓テスト整備とセットで本番デプロイまで自動化し、失敗時の切り戻しも設計
✕運用が属人化し、失敗時のリカバリ設計がない	✕切り替え時に引き継げず、資産が残らない	✓運用開始後も定例で改善提案を続け、実行時間・コストを継続的に最適化
✕CI の実行時間が伸びても、改善されず放置される	✕CI 実行コストの最適化や運用開始後の改善までは含まれない	✓既存スタックを尊重し、段階導入で進める
		✓CI 実行時間・分単価の現状診断から始める
※ 運用知見が個人に溜まり、担当者が抜けると誰も触れなくなります。	※ 運用手順書も別契約になりがちです。	※ 運用手順書を納品物として残し、運用フェーズも継続支援できます。

出典： 当社サービスページ「CI/CD構築」（https://shannon.co.jp）の比較整理より

「入れた」ではなく「回っている」がゴール。次ページで、その状態に到達するまでの進め方をご説明します。

Copyright © 2026 SHANNON LLC. All rights reserved.

いきなり全リポジトリへ展開せず、1リポジトリ → 検証 → 横展開で進めます

構築までの流れ。診断から始めて、開発者が自分で触れる状態で納品します。

無料相談 オンライン30分	現状診断	設計・提案 書面で提示	段階的実装	運用支援
・既存リポジトリの状態を伺う ・目指すリリース頻度を確認	・リポジトリ・テスト資産・デプロイ手順を棚卸し ・CI 実行時間・分単価の現状把握 ・改善優先度を整理	・CI/CDツールの選定 ・パイプライン設計 ・移行計画（**並行稼働・切り戻し手順**）	**★ 本番反映** ・テスト → ビルド → デプロイの順で実装 ・差分検証・互換性確認 ・**開発者が触れる状態で納品**	・リリース運用が軌道に乗るまで、定例で改善提案を継続 ・失敗解析・パイプライン調整

1リポジトリで構築 → 検証 → 全リポジトリへ横展開

お約束：既存スタックを尊重（「GitHub Actions に統一しないと駄目」とは言いません）／段階導入／運用手順書を納品物として残し、以後の引き継ぎ・障害対応を社内で完結できるようにします。

Copyright © 2026 SHANNON LLC. All rights reserved.

診断のみ・構築・継続改善

案件規模に応じて3つの形態から選べます

診断だけで終了する選択肢も用意しています。まず現状を可視化してから判断いただけます。

現状診断

- 既存 CI / デプロイ手順の棚卸し
- 改善優先度マップ
- 診断のみで終了する選択肢
- レポート PDF 納品

リポジトリ・テスト資産・デプロイ手順を棚卸しし、改善優先度を整理。まず現状を可視化してから判断いただけます。

BUILD

構築

- テスト・ビルド・デプロイ自動化
- マルチ環境対応
- 本番反映前の検証
- 運用手順書の整備

テスト → ビルド → デプロイの順で段階的に実装し、開発者が自分で触れる状態で納品します。

継続改善

- リリースサイクル運用支援
- 失敗解析・パイプライン調整
- CI コスト最適化
- 対応範囲は契約書面で個別合意

リリース運用が軌道に乗るまで、定例で改善提案を継続。月次 CI コストレポートにも対応します。

※ 料金は要件確定後に書面でご提示します。PHP / Laravel / Go / Node.js / Python / Ruby on Rails / Java / .NET など主要スタックに対応します。

小規模チームでも、リリース頻度・障害削減・属人化解消のいずれかに価値が出るケースが多いです。費用対効果が出にくいと判断される場合は、診断のみで止める選択もご提案します。

Copyright © 2026 SHANNON LLC. All rights reserved.

既存 Jenkins の資産も、社内ネットワーク内のビルドも、捨てずに移行できます

移行・コスト・体制について、ご相談の多い質問とシャノンの回答をまとめました。

	よくあるご質問	シャノンの回答
移行	既存の Jenkins を捨てる必要はありますか？	捨てる必要はありません。Jenkins のパイプライン資産を読み解き、並行稼働期間を設けて段階的に移行する計画を組みます。
	社内ネットワーク内のビルドにも対応できますか？	対応します。セルフホストランナーを社内に立てる構成や、AWS CodeBuild の VPC 内実行など、ネットワーク要件にあわせて設計します。
コスト	CI コストが膨らんでいて困っています。	コスト最適化の単発診断にも対応します。長時間ジョブ・キャッシュ未活用・並列度過多を棚卸しし、削減施策を優先度付きで提案します。
	小規模チームでも費用対効果は出ますか？	3 名以下でも、リリース頻度・障害削減・属人化解消のいずれかに価値が出るケースが多いです。出にくい場合は診断のみの選択もご提案します。
体制	テストが整備されていない状態でも依頼できますか？	対応します。CI 整備とあわせてテストピラミッド設計・テスト追加方針も提案し、「CI を入れたけど通すテストがない」状態を避けます。
	対応スタックは何ですか？	PHP / Laravel / Go / Node.js / Python / Ruby on Rails / Java / .NET など。コンテナ化・Terraform / IaC とあわせた構築経験を持つメンバーが対応します。

出典： 当社サービスページ「CI/CD構築」（https://shannon.co.jp）よくあるご質問より

Copyright © 2026 SHANNON LLC. All rights reserved.

会社概要

社名	シャノン合同会社（SHANNON LIMITED LIABILITY COMPANY）
代表者	三浦 昌大（Masahiro Miura）
設立	2026年4月 （決算月：3月）
拠点	東京都豊島区池袋2丁目17-8 ／ 千葉県柏市若柴178-4 ／ 北海道札幌市中央区北二条東3丁目2番2号マルタビル8階
資本金	9,900,000円
事業内容	プロダクト開発・運営 ／ インフラ構築・運用 ／ エンジニアリングエージェンシー
パートナー	Xserverビジネスパートナー ／ Microsoft Cloud Partner Program
取引銀行	三菱UFJ銀行 柏支店
電話	050-1794-9651 （自動音声でお受けし、営業日に折り返しご連絡します）
連絡先	contact@shannon.co.jp ／ https://shannon.co.jp

Copyright © 2026 SHANNON LLC. All rights reserved.

E.O.F

SHANNON