

動きのあるWebサイトを 「壊さず」作る

Next.js × GSAP アニメーション実装ガイド 第1版

対象読者：自社サイトに動き・演出を取り入れたいWeb担当者、
制作会社に発注・レビューする立場の方

エグゼクティブサマリー（結論）

- 1 **動き（アニメーション）は今もブランド差別化に有効だが、実装を誤ると「壊れる／重くなる／取り残される」という3種類の事故を起こす。**
- 2 特にNext.js App Routerのような最新フレームワークには、**公式ドキュメントに明記されていない構造的な罣**があり、これを知らずに実装すると「ページ内では動くのに、リンクをクリックして遷移すると動かなくなる」という発見しにくいバグを埋め込む。
- 3 **写真素材が少ない・使えない場合でも、追加の有料アセットなしでリッチな視覚表現は作れる。**
- 4 本書は、自社コーポレートサイト（co-linkweb.com、Next.js App Router＋GSAP/ScrollTrigger構成）の構築過程で得た知見を、GSAP公式・Google web.dev・Awwwards等の一次情報と突き合わせて整理した、**次に同種のサイトを作る際にそのまま使える実装ガイド**である。

背景

企業サイトにおける「動き」は、単なる装飾ではなく、第一印象・専門性の伝達・競合との差別化に直結する要素として重視されるようになってきている。一方で、動きを取り入れたサイトの多くが次のいずれかの事故を起こす。

事故の型	内容
重くなる事故	スクロールのたびにガクつく、モバイルで極端に遅い
壊れる事故	一部のブラウザ・端末で表示が崩れる、ページ内遷移後に演出が発生しない
取り残す事故	めまい・吐き気などを誘発しうる動きを、それを望まない訪問者にも一律に見せてしまう

本書は、Next.js（App Router）とGSAP（ScrollTrigger）を用いた実装を主な対象に、これらの事故を構造的に防ぐための原則・手順・チェックリストをまとめたものである。

事実（一次情報に基づく知見）

1. 演出強度は「見るもの／利用するもの」で分岐させる

ページの性質	該当例	演出強度
見るもの（ブランド訴求・初回接触）	トップページ、キャンペーンLP	開いた瞬間の大きな演出を許容してよい
利用するもの（実務ツール・情報探索）	サービス詳細、料金表、フォーム	演出は控えめに。常時アニメーション・自動再生ループは避ける

この判定を誤ると、「情報を探しに来た人が演出を待たされる」「第一印象を作るべきページが静的すぎる」という、方向の異なる2つの失敗が起きる。

2. Next.js App Router最大の落とし穴：共有レイアウトの初期化漏れ

Next.js App Routerでは、共有レイアウト（`layout.tsx`）に置いたクライアントコンポーネントは、クライアント内遷移（リンククリック）では再マウントされない。DOM要素をスキャン・監視する初期化处理（依存配列が空の`useEffect`）は、アプリ起動後の最初の1回しか実行されず、遷移先ページのDOMには一度も適用されない。

これはGSAP公式ドキュメントに明記があるわけではなく、Next.js公式の`usePathname`ドキュメントの文脈やGSAPコミュニティフォーラムで繰り返し言及される、**フレームワーク特有の構造的な罠**である。URL直打ちでは再現しないため、通常の動作確認だけでは見逃しやすい。

対処の要点：DOM要素を収集・監視するクライアントコンポーネントをレイアウトに置く場合、`usePathname()`を取得し、`useEffect`の依存配列に含める（クリーンアップで前回分のオブザーバーを確実に破棄してから再構築する）。動作確認は「URL直打ちで表示されるか」だけでなく、**実際にナビゲーションリンクをクリックして遷移した状態でも必ず行う**。

3. パフォーマンスの規律

- アニメーションは`transform`/`opacity`のみを対象にする（レイアウトを引き起こすプロパティは避ける。Chrome for Developers公式は、これらをアニメーションするページは平均よりCLS「良好」達成率が低いというデータを示している）
- `will-change`は「向こう200ミリ秒以内に動く要素」だけに限定使用する。乱用するとメモリ消費が増え、かえって遅くなる
- 大量の要素を演出する場合はGSAPの`ScrollTrigger.batch()`（グループ化・段階的初期化）を使う

4. アクセシビリティの規律

すべてのアニメーションコンポーネントは、OS設定`prefers-reduced-motion: reduce`を読み取り、対応する分岐を持つべきである（web.dev・MDN公式）。前庭障害（めまい・吐き気）のリスクが高い大きな動き（パララックス、自動ループ演出）は、この設定が有効な場合に省略・短縮する。完全排除ではなく「必須でない動きを減らす」対応でよい。

5. パフォーマンス予算（Core Web Vitals・Google公式基準）

指標	目安
LCP (Largest Contentful Paint)	2.5秒以内
INP (Interaction to Next Paint)	200ミリ秒以内
CLS (Cumulative Layout Shift)	0.1以内

いずれか未達の場合は、演出の尺短縮やパーティクル数の削減などで調整してから公開することを推奨する。

解釈

上記の落とし穴（特に2番目のNext.js App Router特有の罠）が見過ごされやすい理由は、①ライブラリ公式ドキュメントに明記がない、②URL直打ちの確認では再現しない、という2点による。制作会社に発注する場合、納品前の確認項目に「リンククリックによる内部遷移後もアニメーションが発火するか」を明示的に含めない限り、検収時に見逃される可能性が高い。

同様に、「オシャレだが重い」「動くが一部の人には配慮がない」という失敗の多くは、技術力不足というより、**チェックすべき項目がそもそもチェックリスト化されていないこと**に起因する。

選択肢

選択肢	内容	向くケース
A：自前実装する	本書の手順・チェックリストに沿って内製または発注管理する	差別化を重視し、中長期でサイトを育てる予定がある
B：条件付きで採用する	ブランド訴求ページのみ演出を厚くし、実務ページは控えめにする	リソースが限られており、効果が見込める箇所に絞りたい
C：動きを入れない	演出を持たず、情報設計とパフォーマンスに集中する	更新頻度が高い純粋な業務ツール・社内向けサイト等

どれが適切かは事業内容・更新体制・予算による。本書は選択そのものを代行しない。

実行案（フェーズ）

- Phase 0 ページ一覧を作り、「見るもの／利用するもの」に仕分ける
- Phase 1 技術スタックを決める（本書はNext.js App Router+GSAP+ScrollTriggerを既定値として扱う）
- Phase 2 レイアウト常駐コンポーネントの初期化漏れ対策を実装規律として徹底する
- Phase 3 写真に頼らない視覚表現（下表）を必要箇所に適用する
- Phase 4 単調化チェック（同じレイアウトの連続を避ける）
- Phase 5 公開前QA（次ページのチェックリスト）

写真に頼らないリッチな視覚表現カタログ（追加アセット不要）

技法	視覚効果
ノードネットワーク（Canvas 2D）	点と線が有機的に繋がる知的な背景
3Dパーティクル（Three.js）	奥行きのある発光する粒子群
静かな光の明滅（CSSのみ）	WebGL不要の軽量な”呼吸する”光
線描画アニメーション（SVG）	パスが手描きのように伸びていく
リング・オービット演出	ブランドページの大演出に向く

成果物（本書を使うと手元に残るもの）

ページ分類チェックシート／Next.js App Router特有の罨に対する実装チェック項目／写真不要の視覚表現カタログ／公開前QAチェックリスト（下記）

公開前QAチェックリスト

- ☐ レイアウト常駐コンポーネントが、内部遷移のたびに正しく再初期化されるか
- ☐ クリックによるページ内遷移で、全アニメーションが正しく発火するか（URL直打ちの確認だけで済ませない）
- ☐ prefers-reduced-motion: reduce 環境で過剰な動きが抑制されるか
- ☐ transform／opacity以外のプロパティをアニメーションしていないか
- ☐ モバイル実機での表示・タッチ操作を確認したか
- ☐ Core Web Vitals（LCP／INP／CLS）が目標値内か
- ☐ WebGL初期化に失敗した場合の代替表示（フォールバック）を用意しているか

リスク・限界（正直な注記）

本書の一部の記述（例：「動きを止めた静止画でも成立する強さを持たせる」等）は、業界の経験則レベルであり、厳密な実証データがあるわけではない。該当箇所は本文中に「経験則」等の言葉で明示している。

WebGL非対応環境・低スペック端末への配慮（フォールバック設計）は個別に検証が必要である。

本書は一般的な技術解説であり、個別サイトでの実装結果を保証するものではない。ブラウザ・端末の組み合わせによって挙動が異なる場合があるため、自社サイトへの適用時は必ず実機で検証すること。

次の一步

本書の内容は、実装のたたき台として自由に利用してよい。自社での実装体制がない、またはNext.js App Router特有の罨を踏まずに一気に作り切りたい場合は、CO-LINK（AI導入支援・マーケティング事業）までご相談いただきたい。

株式会社CO-LINK info@co-link1618.com

付録

用語集

GSAP / ScrollTrigger

JavaScriptのアニメーションライブラリ、およびスクロール連動を扱う拡張機能

CLS (Cumulative Layout Shift)

ページ表示中に要素がどれだけ予期せず動くかを示す指標

INP (Interaction to Next Paint)

ユーザー操作への応答速度を示す指標

LCP (Largest Contentful Paint)

メインコンテンツの表示完了までの時間を示す指標

prefers-reduced-motion

OS側の「視差効果や動きを減らす」設定をWebサイト側で検知するための仕組み

出典一覧

- GSAP公式「React & GSAP」 (gsap.com/resources/React/)
- GSAP公式 `gsap.matchMedia()` ドキュメント
- Next.js公式 `usePathname` ドキュメント
- web.dev・MDN公式 `prefers-reduced-motion`
- web.dev公式「Animations and performance」
- Chrome for Developers公式「Avoid non-composited animations」
- Awwwards公式「Evaluation System」
- CSS-Tricks「How SVG Line Animation Works」
- Smashing Magazine「Magical SVG Techniques」

いずれも2026-09-02時点で一次情報を確認済み。URL・仕様は変更され得るため、実装時は最新版を都度確認すること。

版管理

版	日付	内容
v1.0	2026-09-15	初版発行

© 2026 株式会社CO-LINK. All rights reserved. 無断複製・転載を禁止しますが、本書内容を参考に自社サイトの改善に役立てることは自由です。