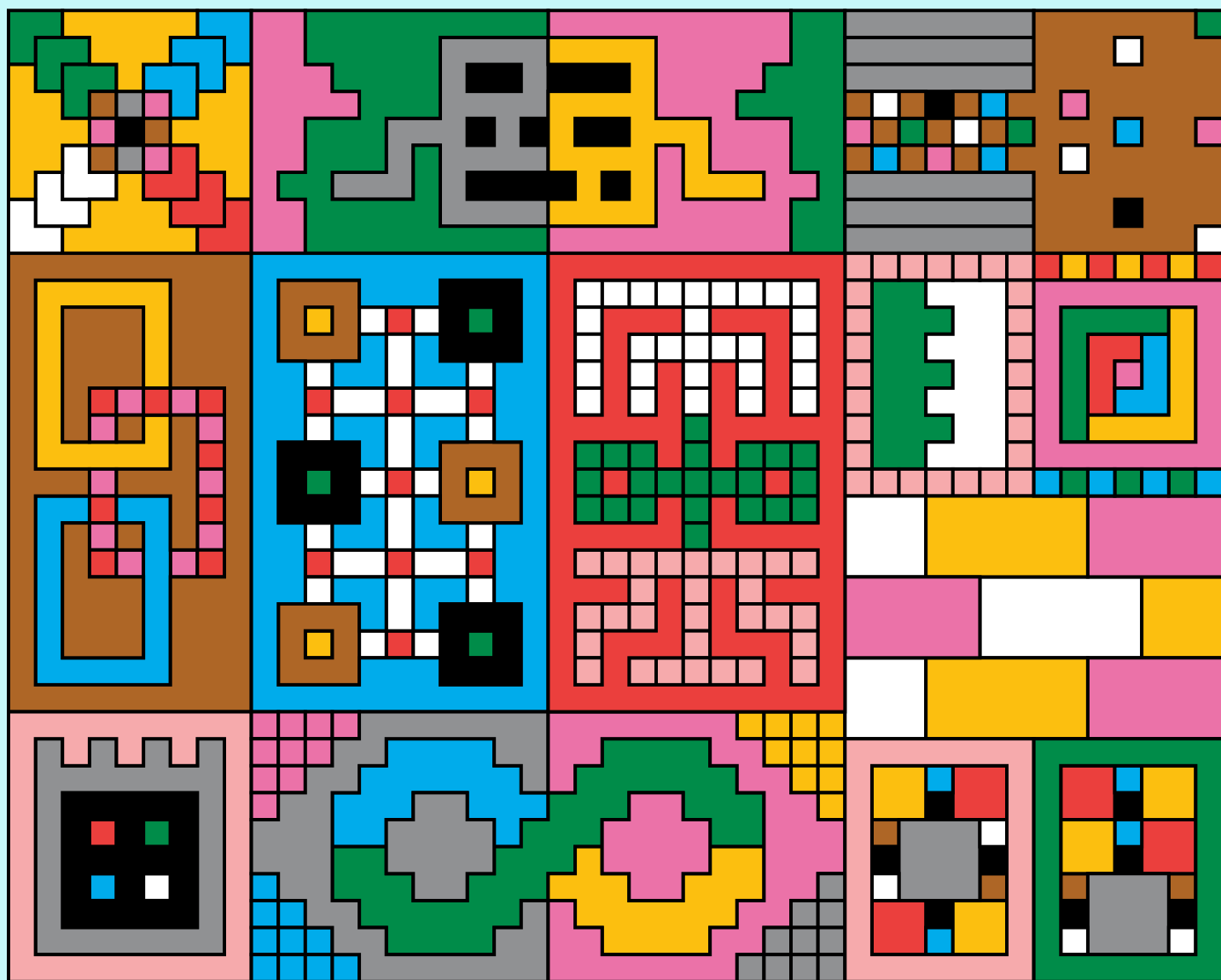




AI時代の 日本の開発者向け ガイドブック



今日、日本の開発チームは、APAC全体と比較してAIの使用がそれほど広範囲には及んでいません。APACの開発者調査によると、過去12ヶ月間に全てのプロジェクトでAIを使用したと回答した日本の開発者は19%にとどまり、APAC全体の23%に比べて低い割合です。多くの日本の開発者は、AIでより生産的になることができると考えていますが、一方で49%は迅速なリリースには役立っていないと答えており、APACの58%の開発者が迅速なリリースに役立つと答えているのとは対照的です。

日本の開発者の中で迅速にリリースできないと答えたうちの31%は、使用しているAIツールには、正確なアウトプットを生成するためのデザインコンテキストが不足していると述べ、41%はAIのアウトプットの手戻りが遅れの原因だとしています。しかし、あらかじめ組織に文書化された正式なデザインシステムがあると回答したのはわずか51%で、38%は文書化されていない基盤から作業している、11%はデザインシステムがまったく整備されていないと答えています。

では、日本の開発者はAIをうまく活用するために、何を変えればよいのでしょうか。必要なのは、チームのAIガバナンスの向上と、ツールに与えるデザインコンテキストの改善です。ここでは、より早く本番環境に反映するために、AIのアウトプットを安定させる方法を見ていきます。

主な調査結果

- **AIによって、日本のチームの開発速度はまだ上がっていません。** AIを生産性向上のために使用している日本の開発者のうち、49%が速度は向上していないと回答しています。さらに、AIが速度向上に役立っていないと答えた開発者のうち、31%はAIツールに正しいアウトプットを生成するためのデザインコンテキストが不足していると述べ、41%は手戻りに時間を取られています。
- **多くの開発者は、使用するAIツールがデザインシステムを参照できず、手戻りが発生していると指摘しています。** 調査結果では、45%の開発者が、AIツールはデザインシステムを一貫して参照していないと回答しました。また、AIがリリースの高速化に役立っていないと答えた日本の開発者のうち、43%は、AIのアウトプットを本番環境に反映するまでに依然として6~10回の修正が必要だと答えています。
- **日本の開発者チームのうち、正式なデザインシステムを持っているのは半数にとどまります。** 多くのチームには、デザイナー、開発者、AIツールが共通して参照する信頼できる情報源がありません。

数字で見る

705人の開発者を対象にした調査に基づく ※1

納品速度が変わらない

49%

AIが生産性に影響を与えると答えた開発者のうち、納品までの時間が変わっていないと回答

デザインシステムの整備不足

49%

文書化された正式なデザインシステムが存在しない

手戻りの増加

43%

AIのアウトプットに6～10回のプロンプトや修正が必要

参照のばらつき

45%

AIコーディングツールによるデザインシステムの参照に一貫性がないと回答

限定的なコラボレーション

25%

プロダクトデザインに深く関与していると回答。APACで最も低い割合

一貫性のないAI運用

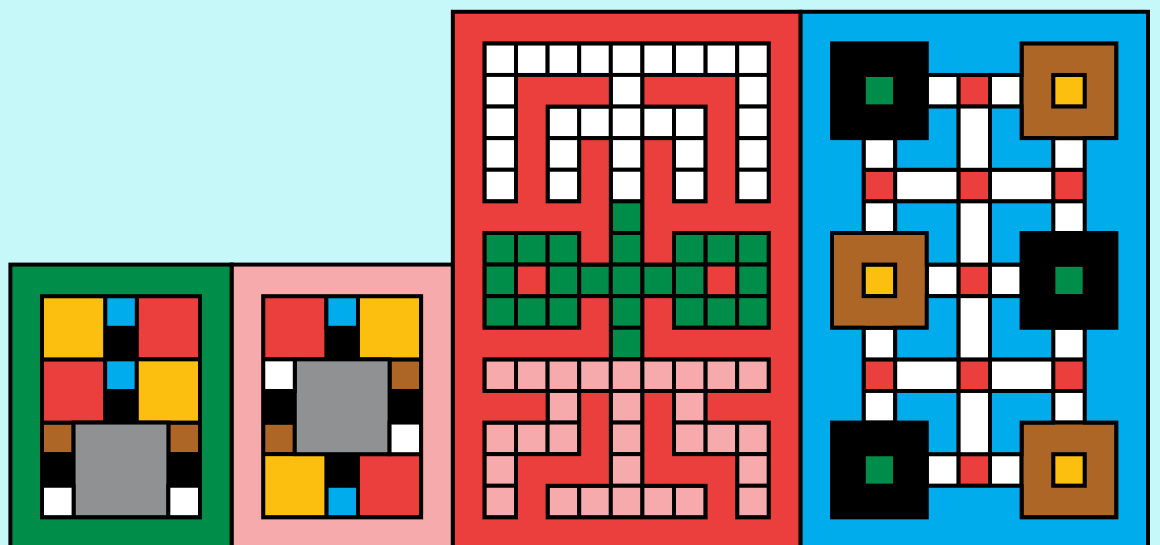
43%

組織にはより明確なAI基準が必要だと感じている

※1 Figmaの委託により、YouGovがオーストラリア、ニュージーランド、シンガポール、日本、韓国の開発者705人を対象に、2026年5月から6月にかけて調査を実施しました。回答者は、従業員10人以上の組織に勤務し、AIコーディングツールを使用し、過去12ヶ月間にAIを活用したプロジェクトを完了し、開発者以外の職種と定期的に連携している人です。

これからの打ち手： より良いAIガバナンス、 連携の取れたワークフ ロー、スケーラブルな デザインシステム

AI対応のデザインシステムであれ、明確なAI基準であれ、コンテキストはチームがAIをより効率的に使用するのに役立ちます。コンテキストがなければ、開発者はAIのアウトプットの手戻りや煩雑なレビューサイクルへの対応に時間を取られてしまいます。ここでは、AIを活用した開発を改善し、成果物をより早く本番環境に反映するための3つの戦術をご紹介します。



1 より良いAIガバナンスとイネーブルメントを確立する

AIを活用した開発を拡大するには、適切なツールやモデルを選ぶだけでは足りません。チームには、AIをどのように使うべきか、どの部分に人間のレビューが必要か、本番環境に反映する前に誰が承認の責任を持つのかについて、明確な指針が必要です。

日本では、開発者が組織からのより良いAIサポートを求めています。レビューと承認プロセスの明確化（51%）、AIの使い方に関するガイドラインの整備（47%）、方針の標準化（43%）、AIに特化したトレーニングとオンボーディング（41%）が必要だと回答しています。

ただし、サポートに必要なのはイネーブルメントだけではありません。コンプライアンスも欠かせません。AIを活用した作業が増加する中で、企業には、AIを安全かつ一貫性を持って使用するための実用的な線引きをチームに示し、機微なデータ、人的チェック、責任の所在などをめぐる不確実性を減らすAIガバナンスの実施が必要となってきます。

51%

レビューと承認プロセスの明確化が必要だと回答

47%

AIの使い方に関するガイドラインの整備を希望

AIガバナンスを実践しやすくする方法

- **AIの使用方法に関する組織全体の明確なガイダンスを公開します。**承認済みのツールとモデル、適切なユースケース、データガバナンス、そして最初のプロンプトから最終レビューまでのどの工程でAIを使うのかを網羅します。
- **AIによる成果物をレビューするための一貫した手順を作成します。**どのアウトプットに人間のレビューが必要か、誰が承認の責任を負うか、本番環境への反映前に満たすべき設計・技術・プライバシー・セキュリティの基準を定義します。
- **こうした基準を、実践的なトレーニングとオンボーディングで定着させます。**承認済みのAIワークフロー、よくあるリスク、判断をエスカレーションすべきタイミング、そして本番環境に反映できる水準のAI活用成果物が実際にどのようなものかを、チームに提示します。

2 デザイナーと開発者のワークフローをつなぐ

デザインと開発のギャップを埋めることで、チームは何を作るべきかをより早く合意し、それを迅速に構築し、コンテキストをより簡単に共有できます。デザイナーがプロダクトデザインプロセスの早い段階で意図を共有すると、開発者は作業が形になる過程で、実現可能性、アクセシビリティ、実装上の制約を指摘できます。それにより、チームはトレードオフを早い段階で把握でき、ビルド中に解決が難しくなる問題を避けられます。

しかし日本では、プロダクトデザイン、プロダクト戦略、ビジネスビジョンに深く関与していると答えた開発者は25%にとどまり、APACで最も低い割合です。

25%

プロダクトデザインに深く関わっていると回答した開発者の割合。APACで最も低い

デザインとコードのギャップを埋める方法

- チームが方向性を固める前に、開発者を初期の検討やデザインレビューに参加させます。そうすることで、技術的制約、アクセシビリティ要件、エッジケースが早い段階で浮かび上がります。
- デザイン、実装メモ、決定事項、フィードバックを1つの共有Figmaワークスペースにまとめ、コンセプトからビルドまで、すべてのチームが同じコンテキストを持てるようにします。
- FigJamでアイデアをすり合わせ、Figma Designで体験を形にし、Figma MCP サーバーでデザイン意図をコードに反映します。レビューのためにキャンバスに戻すまで、途中でコンテキストを失うことはありません。

3 チームの基盤となるデザインシステムを作る

AIの効果は与えられたコンテキスト次第であり、適切に整備されたデザインシステムがあれば、チームは最初から本番環境に反映できる状態に近いコードを生成できます。AIツールがデザインコンテキストにアクセスできないと、開発者はコンポーネントの置き換え、トークンの修正、チームがすでに下した実装上の決定のやり直しに時間を取られることになります。

しかし日本は、文書化された正式なデザインシステムの導入率がAPACで最も低くなっています。日本のチームの約半数（49%）が、そうしたデザインシステムを持っていません。そのうち38%は文書化されていない基盤の上で作業しており、11%はデザインシステムを一切持っていません。

デザインシステムがある場合でも、45%の開発者は、AIコーディングツールがそれを一貫して正確に参照していないと回答しており、それがチームの速度を落としています。AIを使ってもリリースが早くなっていない開発者のうち、31%は、AIツールに正しいアウトプットを得るための十分なデザインコンテキストがないと回答し、41%は手戻りに時間を取られていると回答しています。

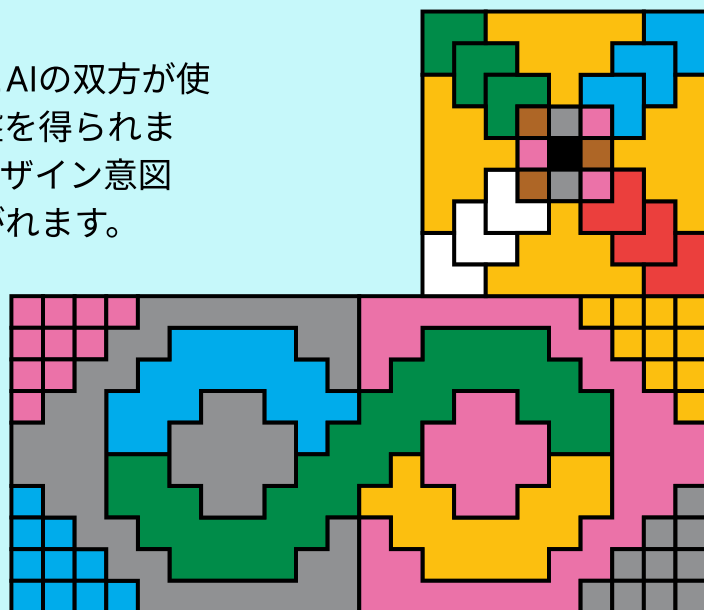
デザインシステムの知識を体系化し、チームとAIの双方が使えるようにすれば、全員がより信頼できる基盤を得られます。承認されたコンポーネント、パターン、デザイン意図が、デザインからコードへスムーズに引き継がれます。

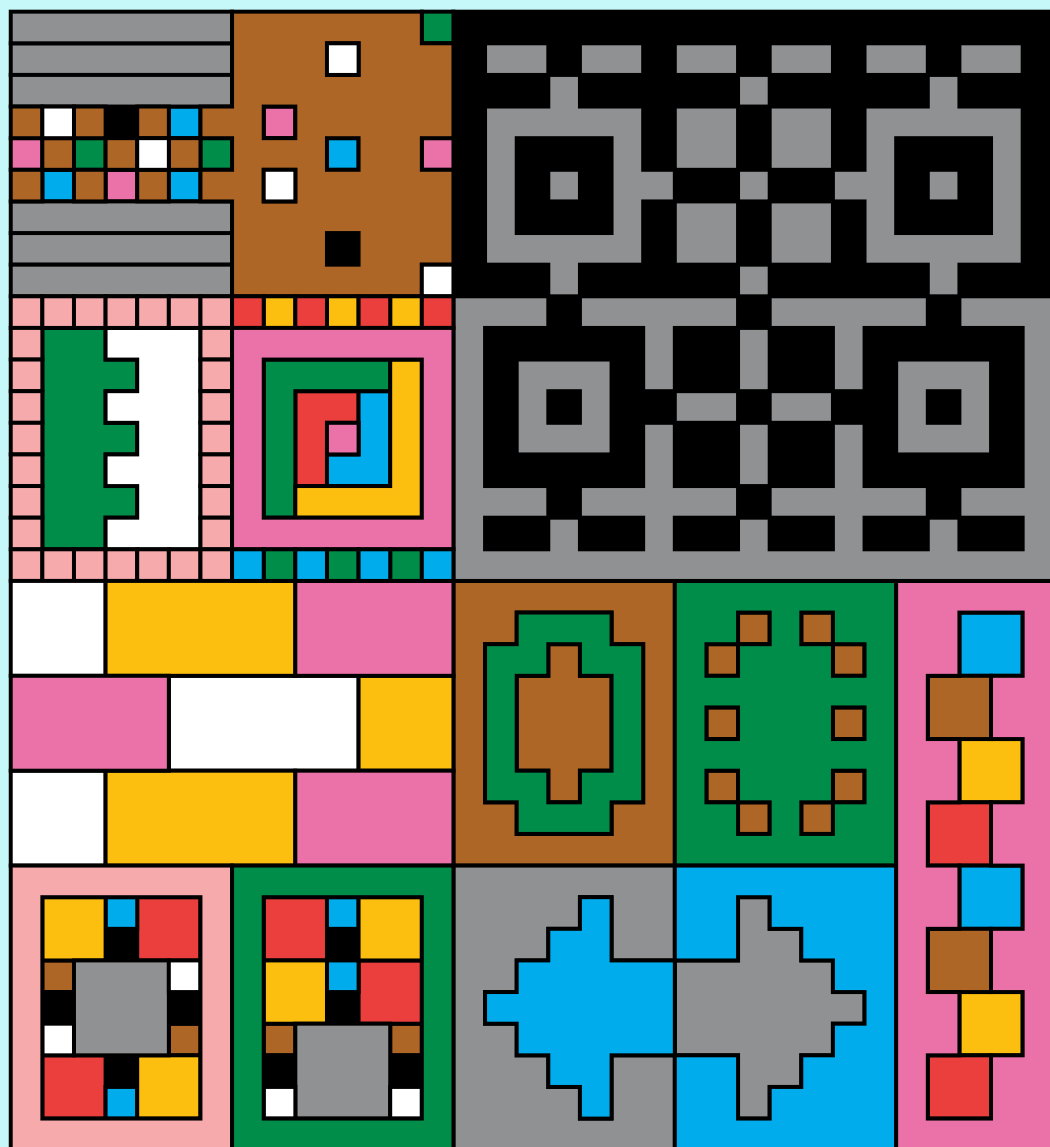
49%

文書化された正式なデザインシステムを持っていない

43%

AIのアウトプットに6~10回のプロンプトや修正が必要





AI対応のデザインシステムを構築する方法

- デザインシステムをしっかりと構造化し、最新かつ明確な状態に保ちます。コンポーネントの目的、使い方、動作、状態、アクセシビリティ要件を文書化し、チームとAIの双方が望ましい状態を理解できるようにします。
- カード、ヘッダー、フォームなどの再利用可能なパターンや上位レベルのブロックを追加し、個々の要素がどう組み合わせるかをAIに推測させるのではなく、AIが土台にできるより完成度の高い構成を渡します。
- Code Connectを使用して承認されたコンポーネントを本番環境側の対応するコンポーネントにマップし、Figma MCPサーバーでそのコンテキストをAIコーディングツールに持ち込みます。生成されたコードが、チームのすでに信頼しているコンポーネント、バリアブル、ドキュメントを参照できるようにします。

Sansan事例：Figmaを使って、AI支援による開発を加速

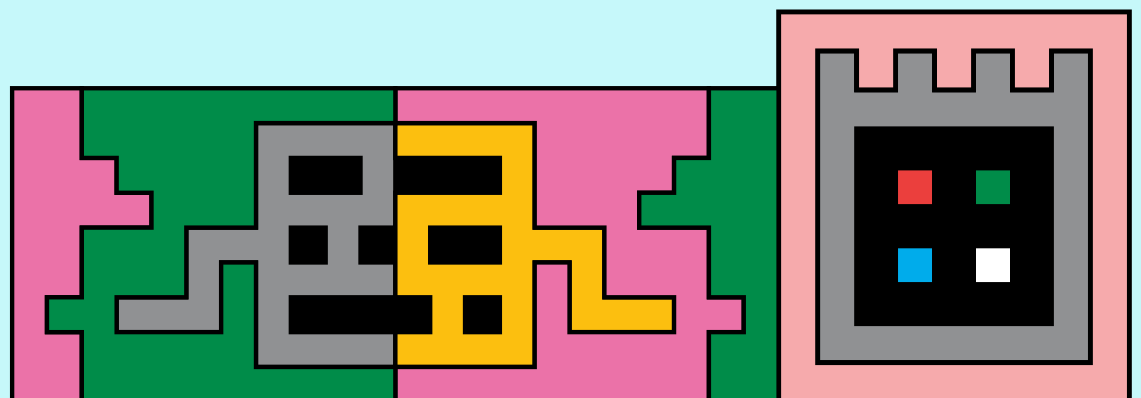
ビジネスデータベース「Sansan」をはじめとしたクラウドベースソリューションを展開するSansanは、AIのデザインからコードへのワークフローを迅速化し、AIのアウトプットでの手戻り時間を短縮したいと考えていました。そこでFigma MCPサーバーとCode Connectを使用して、AIツールがより良いデザインコンテキストにアクセスできるようにしました。

SansanのチームはFigmaのMCPサーバーを利用して、AI契約データベース「Contract One」の新しい画面を作成しました。ワークフローは、開発者が自分のコーディングエージェントにFigma Designファイルへのリンクを提供することから始まりました。FigmaのMCPサーバーを使用すると、エージェントはファイルとSansanのデザインシステムから、特定のコンポーネントやコンテキストを取り込めるようになりました。次に、Figmaコンポーネントをコードベースに直接マッピングするCode Connectを使用して、エージェントは新たにコードを書かずに、Sansanの実際の実装を使用してアウトプットを生成できました。その結果、チームはアウトプットの手戻りに時間を取られることなく、より早く本番環境に進めるようになりました。

FigmaのMCPサーバーとCode Connectによって、コーディングエージェントに重要なデザインコンテキストが提供され、作業時間はおよそ半分に削減されました。

約50%

Figmaでの新規開発プロジェクトにおける時間削減



Figma開発ツールで、デザ
イン、ドキュメント、
コードを一つのワークフ
ローに統合しましょう。

[詳しく見る →](#)

Figma

figma.com/reports