

”本当に使われるシステム”に作り変える

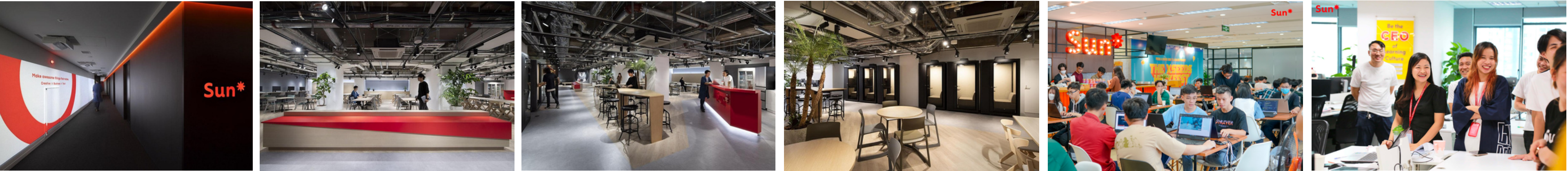
AI×UX モダナイゼーション

Sun*

会社概要

社名	株式会社Sun Asterisk 英語社名 Sun* Inc.	本社所在地	東京都千代田区大手町
設立日	2013年3月1日	事業内容	デジタル・クリエイティブスタジオ事業 4ヶ国6都市に拠点※
従業員数	約 2,080名 グループ全体	資本金等	33億9,419万円 資本準備金含む
関連会社	Sun Asterisk：464 Sun Asterisk Vietnam：1,302 Sun Asterisk Software Development：46 Sun terras：132 NEWh：26 Trys：89 GlobalGear：21 2024年12月末時点（有期雇用等含む）	売上	148.3億円 2025年12月期実績

※日本（東京）ベトナム（ハノイ、ダナン、ホーチミン）＋パートナー拠点 フィリピン（セブ）＋パートナー拠点 カンボジア（プノンペン）



UXのアプローチ

業務システム刷新がDXを牽引する4つの理由

本質的なDXとは、単なるIT導入ではなく、業務の仕組みを根本から見直し、効率化・高度化することにあります。その起点として、業務システムの刷新は企業競争力の鍵を握る重要なステップです。

1 競争力を生む

従来の業務システムを刷新することで、データの可視化やプロセスの自動化が可能になり、迅速な意思決定を支援します。これにより、競争優位性が生まれ、効率的な生産管理や柔軟な対応が可能になります。



2 全体を最適化する

部門間での情報共有やリアルタイムなデータ連携が実現され、製造工程全体の最適化が進みます。在庫管理、納期管理、品質管理など、各業務の効率化が一貫した形で実現されます。



3 持続的な成長を支える

成長に伴い、業務システムが継続的な拡張に対応できるようになることで、企業の持続可能な成長を支えます。未来の変化や新たな技術にも柔軟に対応できる基盤を築けることが、DXの成功に不可欠です。



4 組織文化を変える

システム刷新は、単に技術的な変革だけでなく、企業文化や従業員の働き方にも大きな影響を与えます。DX推進が企業全体に浸透し、デジタル化の意識と技術向上を促進させます。



業務システムが使いにくいと感じる原因

業務システムは、業務効率化の切り札として期待される一方で、その使い勝手の悪さが問題視されています。高額な費用と労力を投じてシステムを導入しても、使い勝手の問題から現場で十分に活用されず、期待した効果が得られないケースも少なくありません。

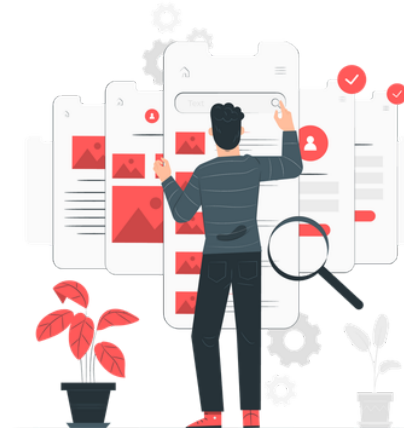
操作性が悪く直感的でない

複雑なUIにより手間や認知負荷が増えると入力ミスや手順漏れが起こりやすく、業務の遅延やトラブルにつながるリスクがあります。



情報過多により画面が複雑

現場では迅速な判断が求められますが、複雑な画面は判断スピードを妨げ、操作ミスや学習コストの増加につながります。



カスタマイズ性の不足

パッケージ製品は自社の業務に合わないことが多く、「業務に合っていない」と感じ、システム活用の意欲が下がる原因になります。



エラーメッセージが不親切。エラーが多い

頻繁に発生するエラーや不具合は、現場にとって大きなストレスで、システムへの信頼や利用意欲の低下を招いています。



システムの動作が遅い。 複数システムの連携が不十分

システムの応答が遅いことや、複数システムの同時利用は、業務全体の効率を大きく下げる要因になります。



現場起点の業務再構築を通じて、 業務システムのモダナイゼーションを推進

変化に強い業務システムへ

現場のユーザー視点で、属人化や非効率的な業務を
棚卸し・再設計し、迷わず使えるUI/UXでシステムを構築。
スクラッチ開発・既存改修・サービス統合まで柔軟に対応
し、環境や要件の変化にも強い、業務システムの設計・
開発を支援します。



Sun*が実践する、ユーザー中心設計に基づいた業務システム開発のアプローチ

業務システムの課題解決の鍵は、徹底したユーザー中心設計（HCD: Human-Centered Design）の考え方を取り入れることです。現場にHCDを適用し、従業員の使いやすさに焦点を当てたアプローチが重要とSun*では考えています。

利用者の業務と ニーズの深い理解



従業員の業務内容やニーズを把握することからはじめます。現場を観察・調査し、ユーザーインタビューを行うことで課題点を洗い出します。現場特有の制約も考慮し、求められている要件を抽出します。

UXリサーチに基づく設計と プロトタイピング



理想のユーザー体験を定義し、プロトタイプを作成します。ユーザーテストで現場の声を拾い、デザインに反映します。こうした検証と改善を反復することで、現場に合ったシステムを開発できます。

管理画面デザインの ベストプラクティス



業務システムのUI/UX検討の際、業務効率を高める体験設計はもちろん、新システムに早く慣れてもらうために学習コストの低い画面を目指し、ユーザビリティやアクセシビリティに配慮した設計を行います。

継続的な改善



業務システムは導入して終わりではありません。継続的に利用者からのフィードバックを収集し、アジャイルに改善を重ねることでPDCAを回し、現場の要望とシステムが提供する価値のズレを最小化します。

開発における、Sun*が重視するポイント

一般的な開発プロセスにおいて、Sun*では徹底したユーザー目線でアプリケーションのUI/UX設計と実装を心掛けることを最大の強みとしています。

	一般的なアプローチ	Sun*のアプローチ
① プロジェクト進行	ウォーターフォール型「要件→設計→実装→テスト→リリース」の順で進行するため、UI/UXは後回しになりがち。	UI/UX起点の開発では、ユーザー体験を軸にプロジェクトを進行。 ユーザー調査やフィードバックをプロジェクト初期から取り入れ、UIとUXの改善を繰り返しながらプロダクトを設計。
② ユーザーの関与	ユーザー観点の確認は要件定義と受け入れテストのタイミングで実施するため、開発途中のフィードバック機会がほぼ無い。	ユーザー調査やユーザビリティテストをプロジェクト全体を通じて継続的に実施 するため、都度ユーザーニーズをキャッチアップ可能。 得たフィードバックをもとに、実用性の高いシステムを設計。
③ 期待値の具現化	初期の要件で設計・開発が進行し、途中で期待値のすり合わせがされないことが多く、納品時にギャップが生じやすい。	ユーザーニーズ・インサイトから、理想とする業務フローを描き、画面設計に落とし込む。 デザイナーがシステム設計の上流から関わることで、ユーザー目線で直感的に使える業務システムを提供。

開発における、Sun*が重視するポイント

一般的な開発プロセスにおいて、Sun*では徹底したユーザー目線でアプリケーションのUI/UX設計と実装を心掛けることを最大の強みとしています。

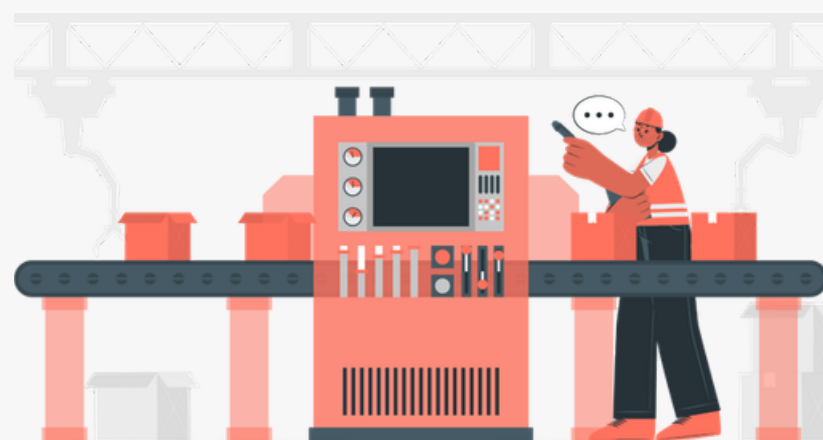
	一般的なアプローチ	Sun*のアプローチ
④ リスク管理	機能や移管に関するリスクは洗い出すが、現場のユーザー視点が抜け漏れ、あとから顕在化することが多く、変更コストが発生する。	ユーザーフィードバックを取り入れながら画面設計を随時見直すため、実装完了後に大きな課題が浮き彫りになるリスクを抑えられる。仕様変更に伴うコスト増やプロジェクトの遅延といった問題も起こりにくくなる。
⑤ 成果物の品質担保	機能テスト・結合テストが中心のため、UI/UXや使いやすさの評価は後回しになったり、スコープ外とされることがある。	ユーザビリティやアクセシビリティに配慮した設計により、リニューアルしたシステムでもスムーズに導入・定着が可能。新たな業務フローの導入が業務効率の向上につながり、真に実効性のあるシステムを導入できる。
⑥ スケジュール予算	初期段階でスコープ・予算・納期を固定するため、要件変更などは追加見積・稟議が必須。そのため途中変更の柔軟性は低くなりがち。	要件の変更が発生した場合でも、デザインおよび実装をアジャイルの思想とプロセスに基づいて進行しているため、プロジェクトの状況に応じてスケジュールや予算を柔軟に調整することが可能。

業務システム刷新においてSun*が担う役割

業務の棚卸しと可視化、業務フローの再設計、現場のユーザー視点でのUI/UXの最適化、柔軟な開発対応までを、現場視点と技術の両面からアプローチします。

① 既存業務の棚卸しと再設計

属人化や手作業の業務を洗い出し、
効率性・柔軟性の高い業務フローに
再構築



② 現場視点のUI/UX設計

実際のユーザー行動や利用環境に
根ざしたUI設計で、現場が迷わず
使えるシステムを実現



③ 柔軟な実装スタイル

スクラッチ開発だけでなく、既存
システムの改修や他サービスとの
連携も含め、最適な形で実現



◆要件定義・設計（ユーザー中心設計）

理解

現場社員へのヒアリングや現場訪問を通じて、ユーザーニーズや課題を理解する

実際に現場へ足を運び、社員へのヒアリングや業務観察を通じて、日々の業務の流れや課題の本質を把握します。業務の全体像だけでなく、ツールの使われ方や非効率なオペレーションなど、ユーザーが抱える潜在的なニーズを丁寧に掘り起こすことで、設計に活かせるリアルなインサイトを得ます。

分析



現場ヒアリング

設計

2025/5/16(金) 栗山さんヒアリング

要約

栗山さんの業務フローと製品に関する要約

業務・プロセスの概要
・一部の部門での業務や役割が重複しているが、業務の流れや課題の把握に役立てる。
・業務の重複や役割の重複は、業務の効率化に役立てる。
・業務の重複や役割の重複は、業務の効率化に役立てる。

業務フローの概要
・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。

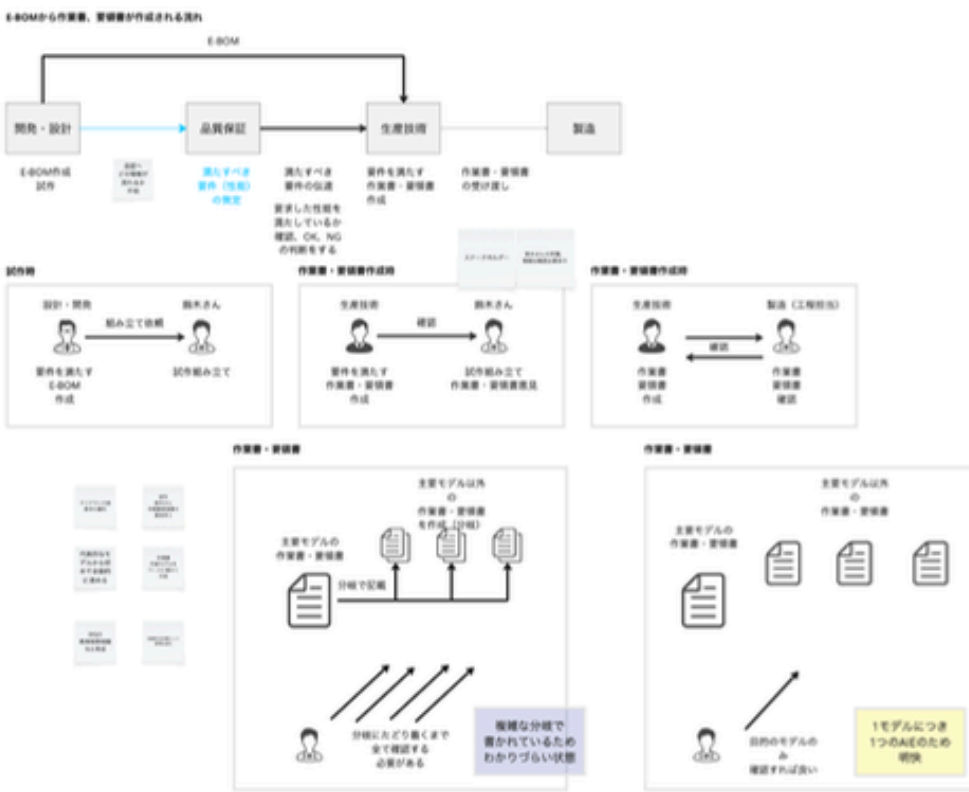
デジタル化の概要
・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。

総括

・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。

ネクストアクション

・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。
・業務の流れや課題の把握に役立てる。



インタビュー結果のすり合わせ

※ 貴社のチームやDX推進フェーズにより、提供サービスの内容や、プロジェクトのアサインする専門家をカスタマイズさせていただきます。

ご支援アプローチの例

◆要件定義・設計（ユーザー中心設計）

理解

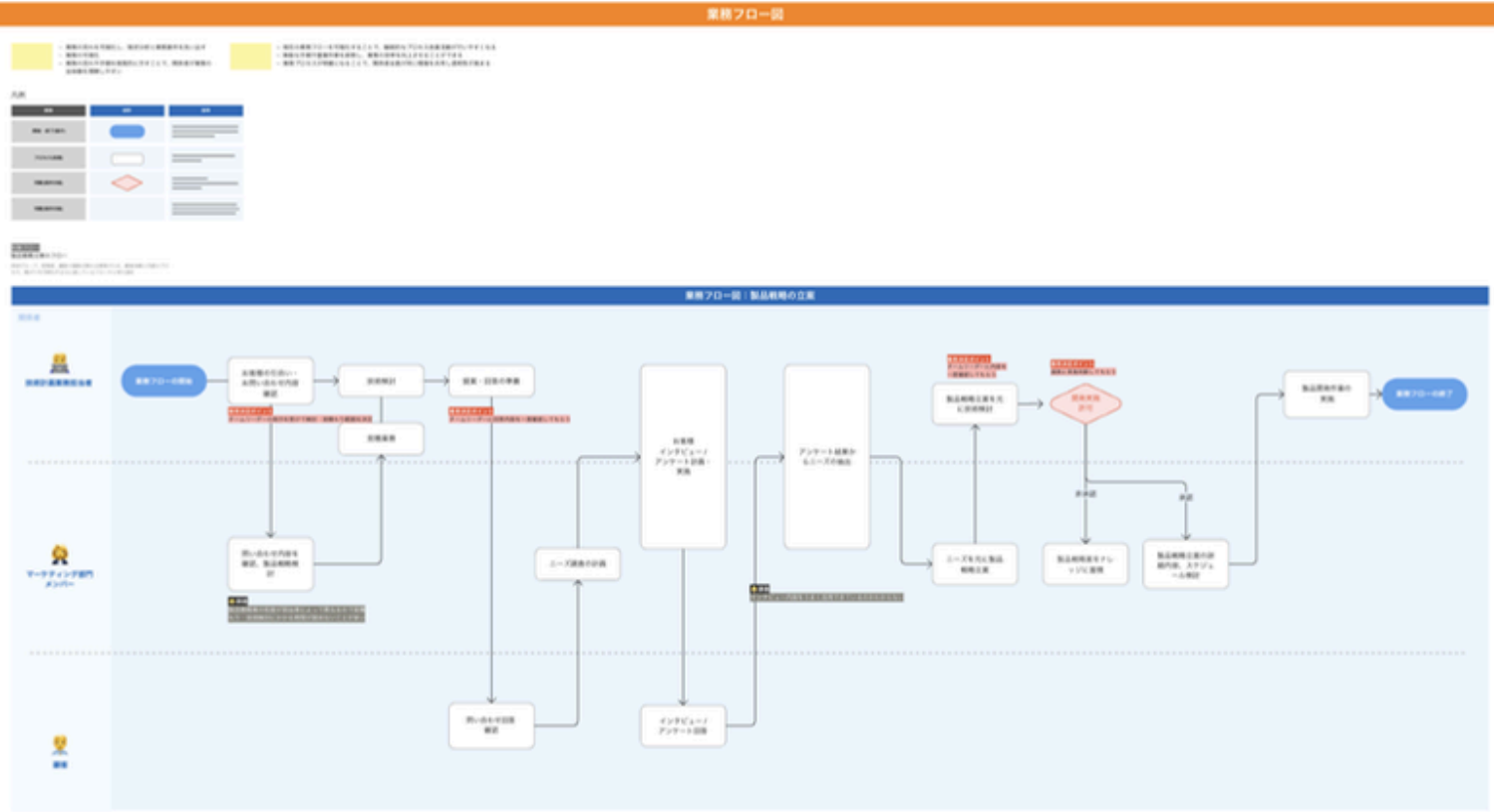
分析

設計

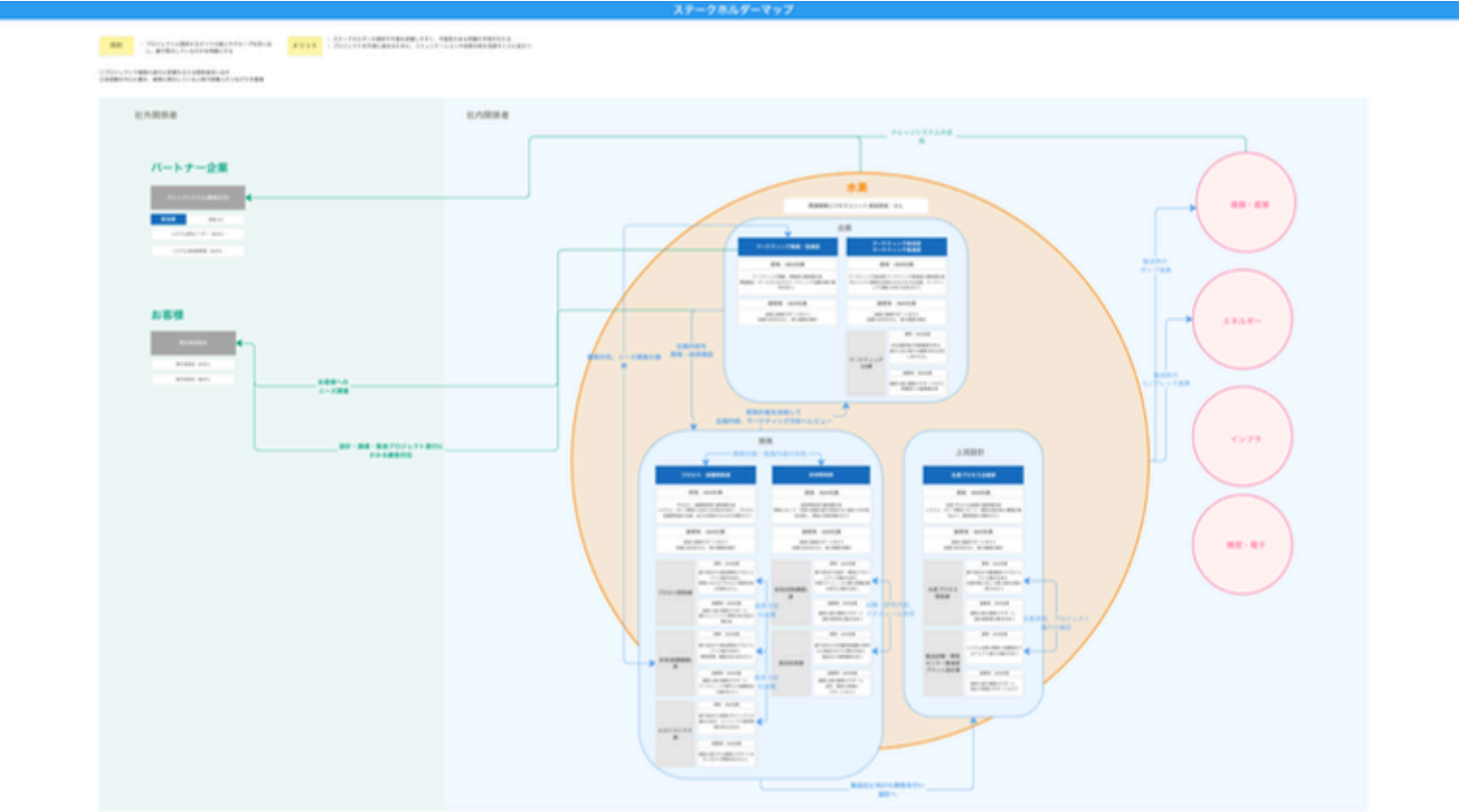
検証

現場リサーチを経て可視化された実態と理想業務とのギャップのある課題を明らかにする

現場調査から得た情報を、業務フロー図やステークホルダーマップとして構造的に可視化。ペルソナの設定や課題の重み付けを行い、理想の業務プロセスとのギャップが大きいポイントを重点的に抽出します。ユーザー体験の設計に直結する本質的な課題を明らかにし、業務システムの設計要求・要件のベースを築きます。



業務フロー図



ステークホルダーマップ

※ 貴社のチームやDX推進フェーズにより、提供サービスの内容や、プロジェクトのアサインする専門家をカスタマイズさせていただきます。

ご支援アプローチの例

◆要件定義・設計（ユーザー中心設計）

理解

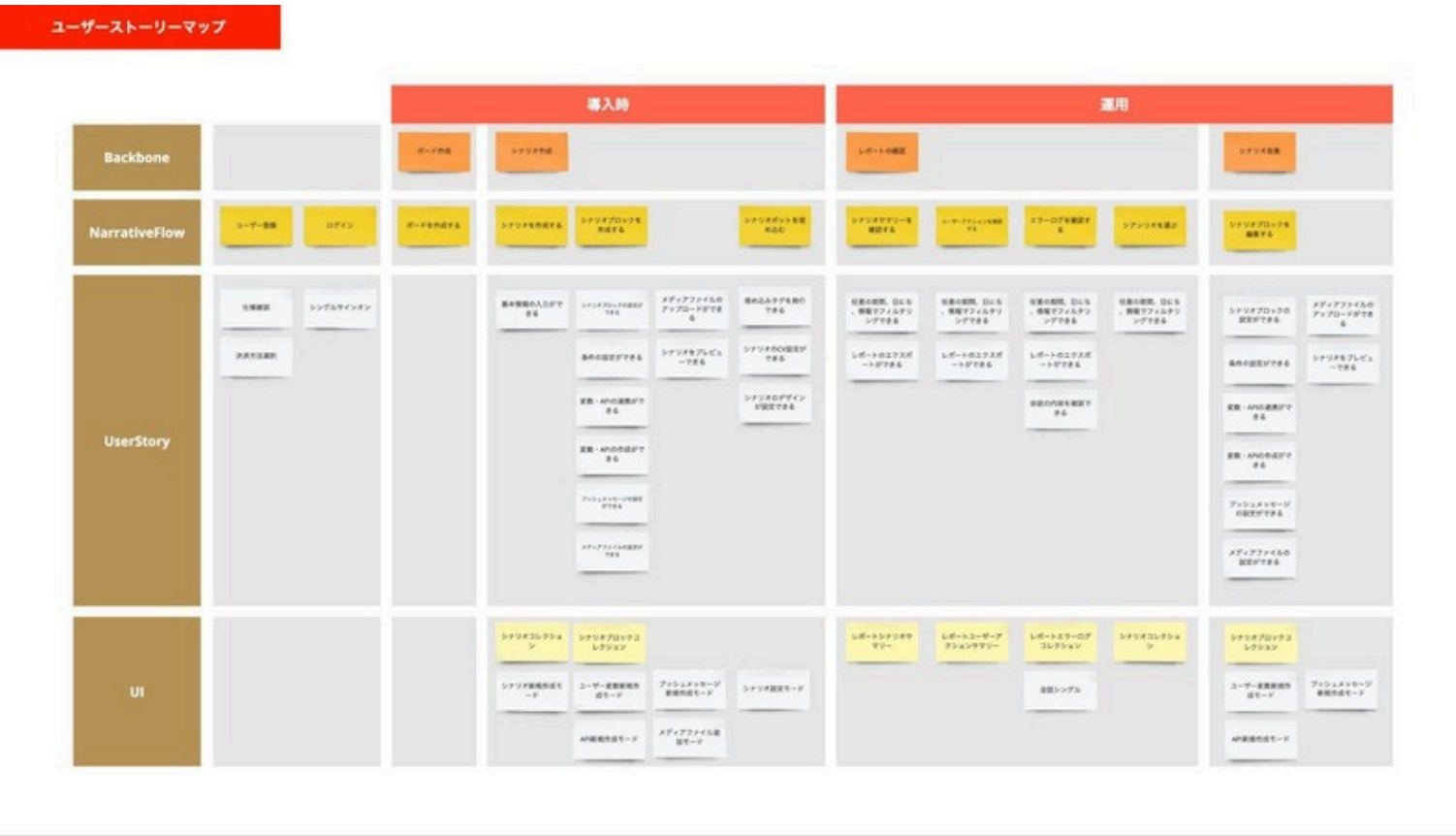
ユーザーストーリーから理想の業務体験を設計し、具体的な画面へと落とし込む

設定したペルソナと業務における課題をもとに、ユーザーストーリーマップを作成。業務シーンごとに必要な機能や情報の流れを整理し、具体的なUIモックアップを作成します。さらに画面同士をつなげたプロトタイプを通じて、業務の流れを視覚的・体感的に再現し、実業務に即した操作性を意識した設計を進めます。

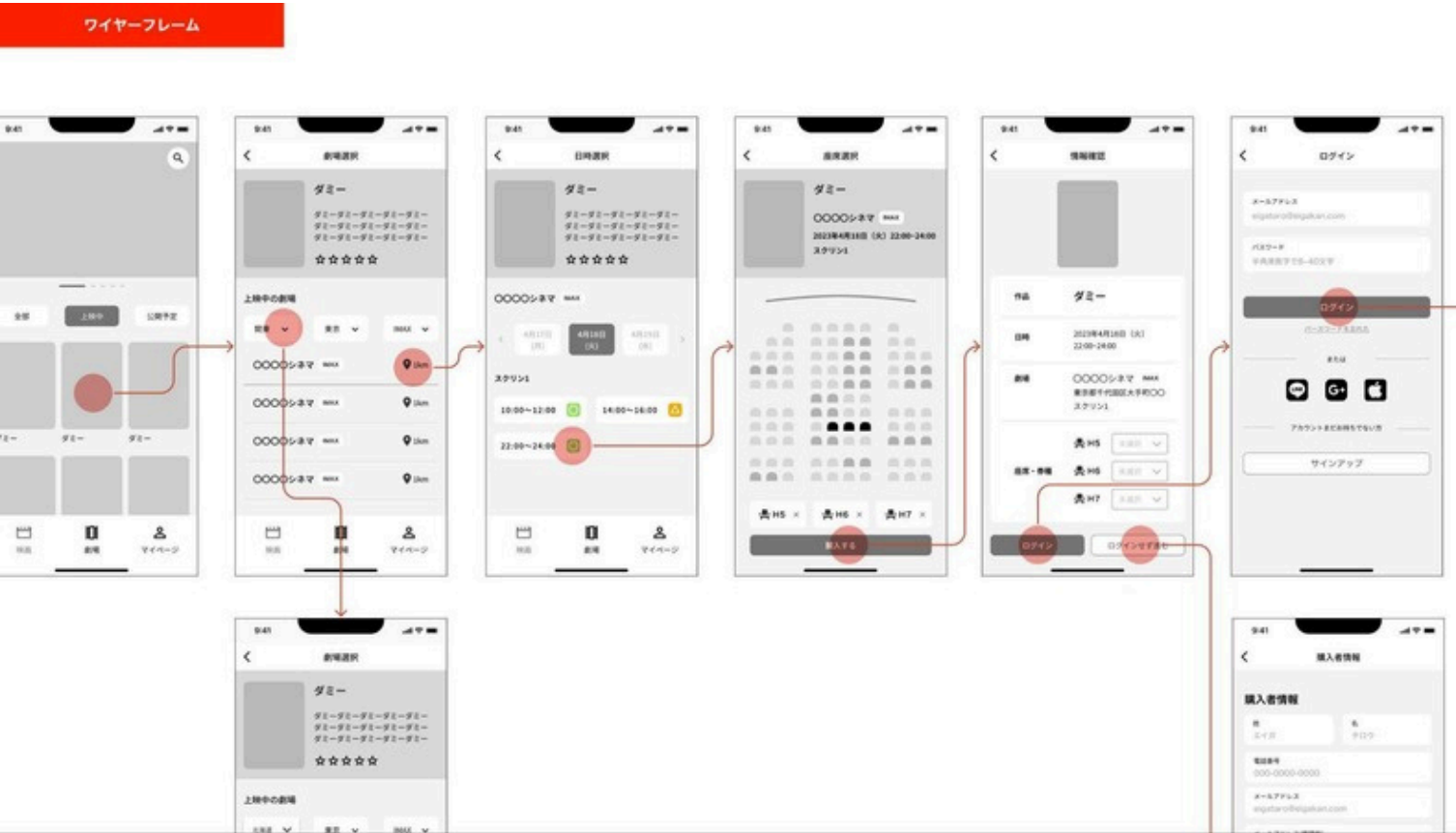
分析

設計

検証



ユーザーストーリーマップ



画面設計

※ 貴社のチームやDX推進フェーズにより、提供サービスの内容や、プロジェクトのアサインする専門家をカスタマイズさせていただきます。

ご支援アプローチの例

- ①既存業務の棚卸しと再設計
- ②現場視点のUI/UX設計
- ③柔軟な実装スタイル

要件定義・設計（ユーザー中心設計）

理解

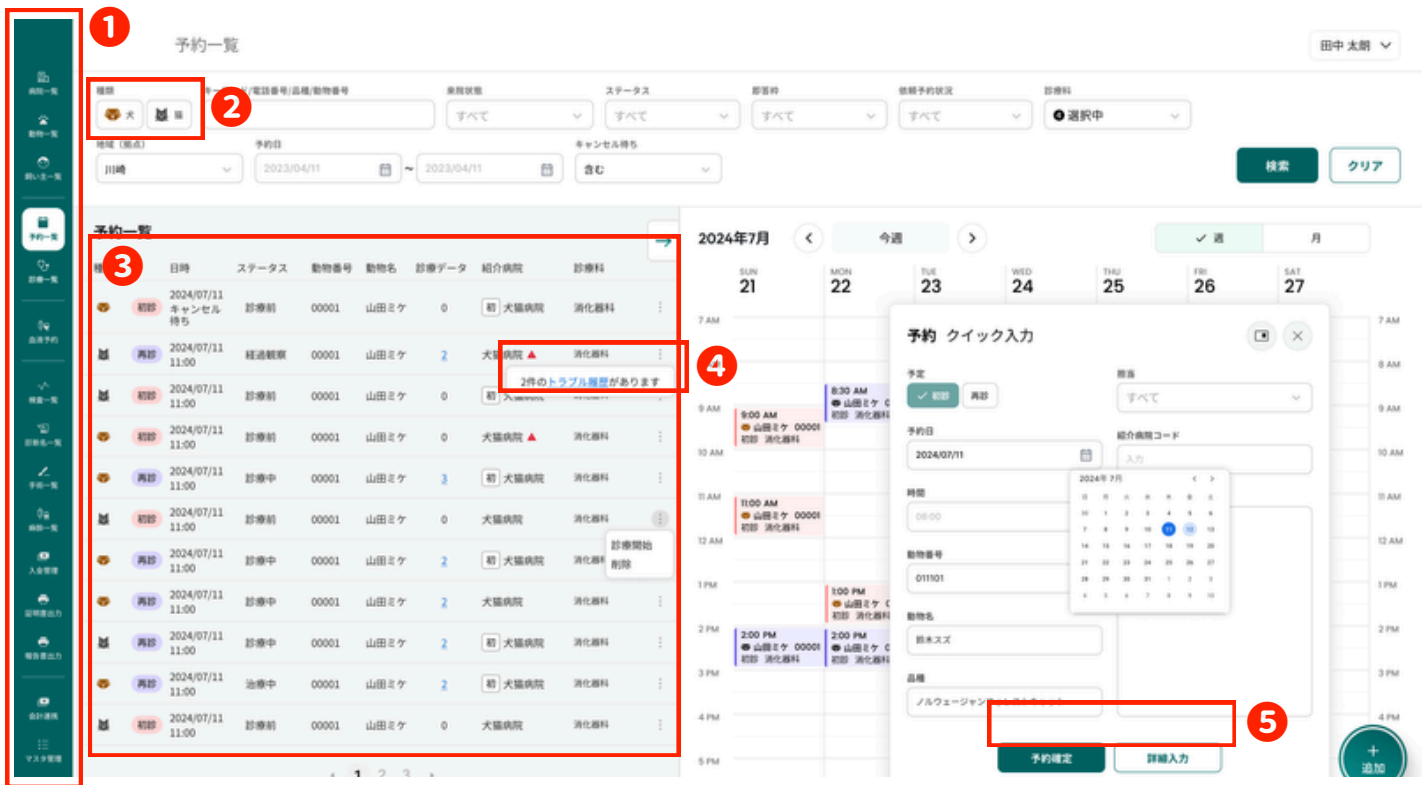
分析

設計

検証

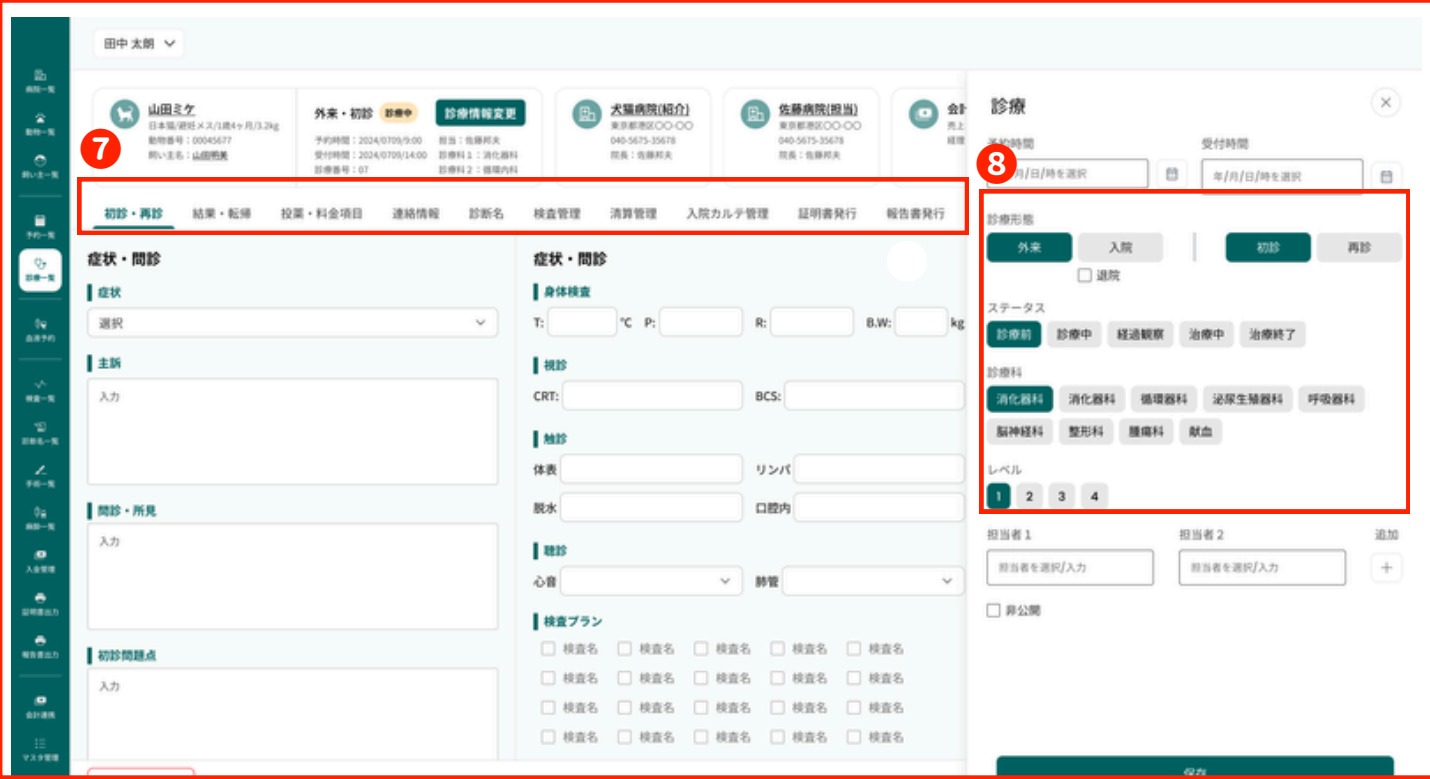
数多くのデジタルプロダクトを手掛けてきた当社が考える、業務システムの画面設計のベストプラクティスを一部ご紹介します。

例：動物病院向け電子カルテシステム



- ① 直感的なナビゲーション、固定エリアはコンパクトに
- ② ショートカットの用意
- ③ テーブル内の情報を見やすくする
- ④ わかりやすいエラーメッセージ
- ⑤ ボタンの優先度をつける、ユーザー視点のボタンラベリング

例：動物病院向け電子カルテシステム



- ⑥ リキッドレイアウトにする
- ⑦ 多層構造は避け、構造をできるだけシンプルにする
- ⑧ 適切に情報をグルーピングして、選択肢を増やしすぎない

ご支援アプローチの例

◆要件定義・設計（ユーザー中心設計）

理解

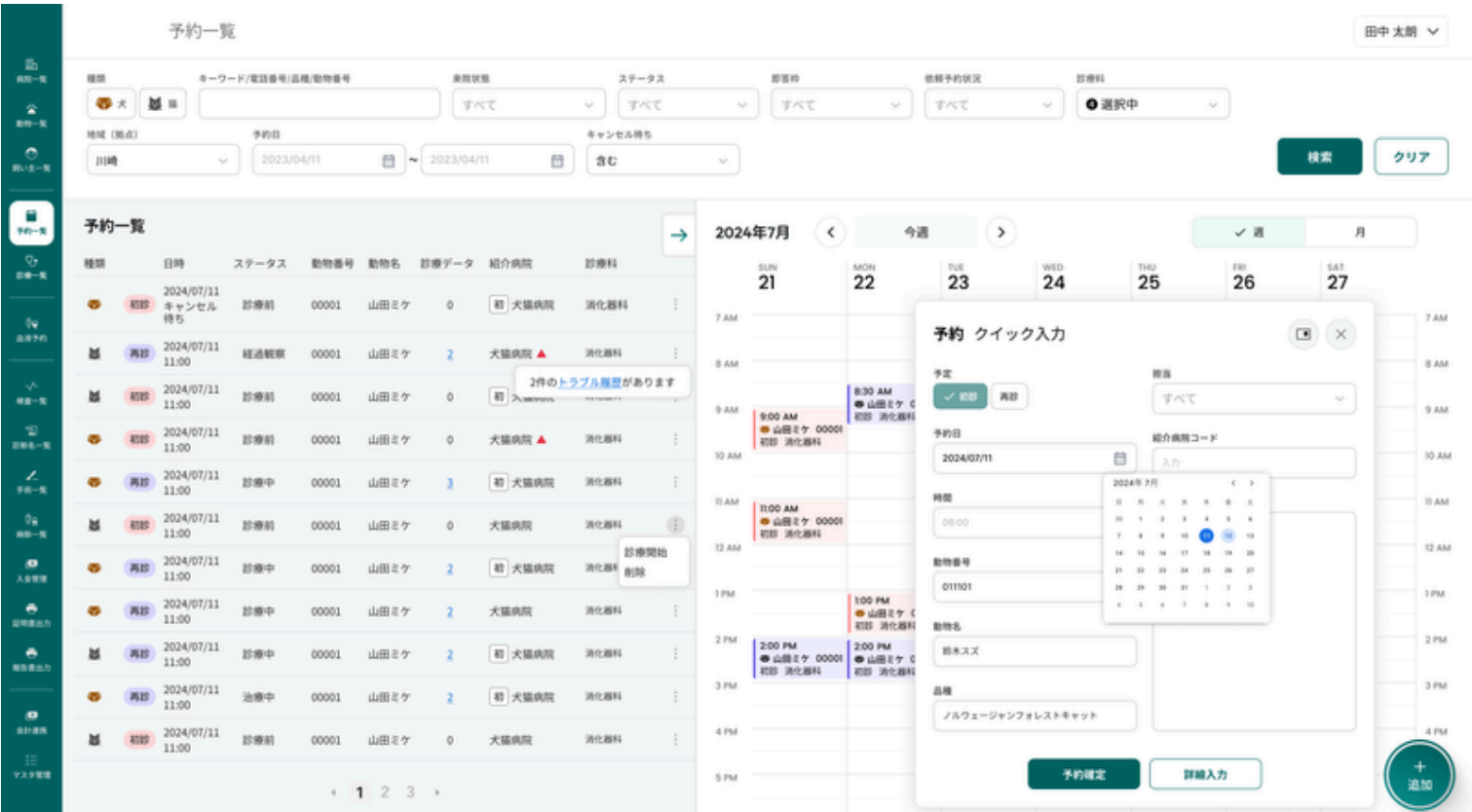
プロトタイプを用いたユーザーテストで、使いやすさを徹底的に検証する

プロトタイプを現場ユーザーに試用してもらい、操作性や理解のしやすさ、業務との適合性などを多面的に評価します。いただいたフィードバックから、改善点や期待に応えるべき要素を抽出し、使いやすさを担保したUI/UXへと磨き上げていきます。開発前の段階で“現場で本当に使われる”プロダクトを目指し、システムの品質を高めていきます。

分析

設計

検証



プロトタイプ



ユーザーテスト

※ 貴社のチームやDX推進フェーズにより、提供サービスの内容や、プロジェクトのアサインする専門家をカスタマイズさせていただきます。

Demonstration

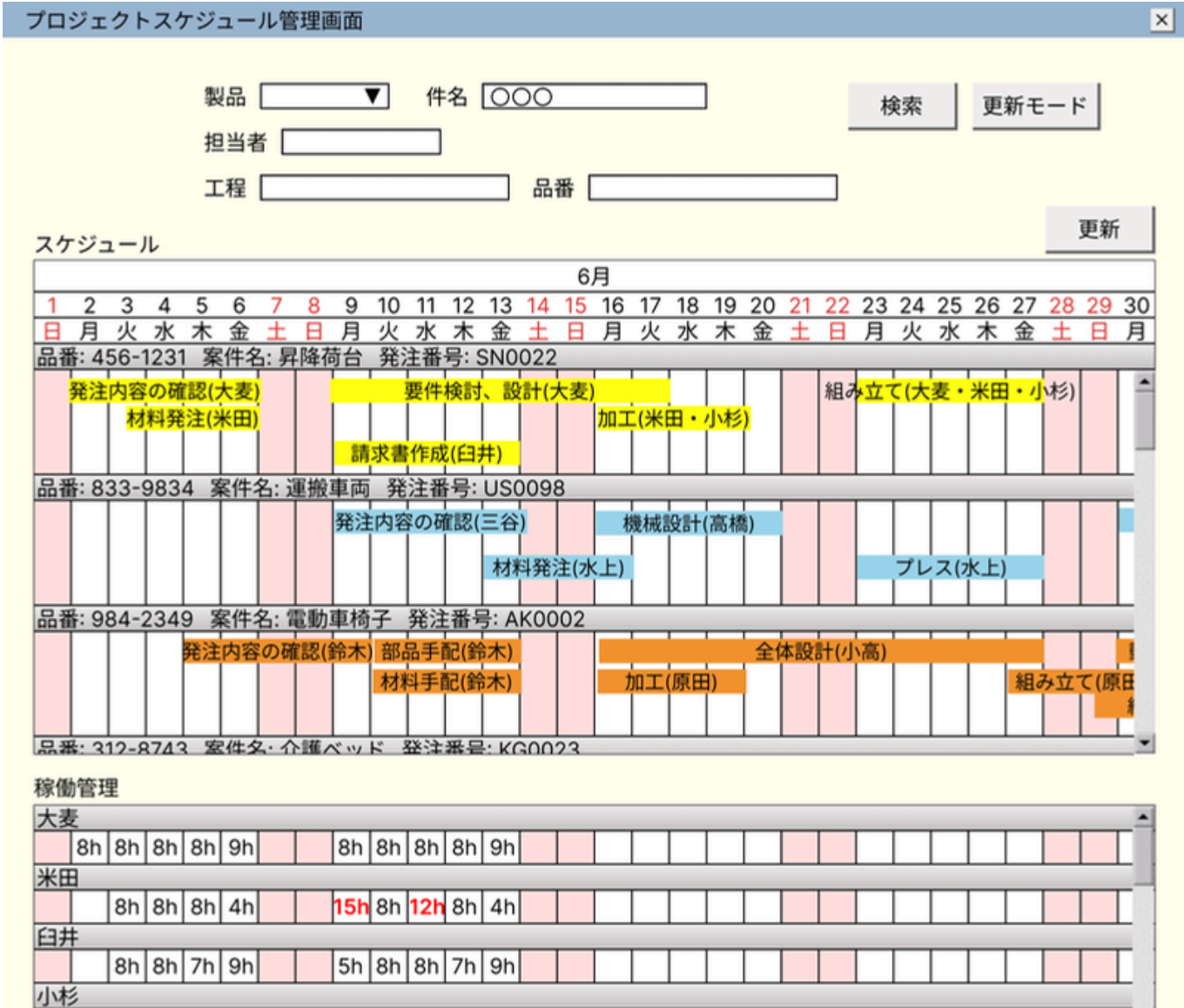
デモンストレーション（生産管理システムの事例）

デモ画面作成の流れ

- 1 いただいた資料をもとに、画面に必要な情報を整理
- 2 ペルソナを整理
- 3 整理した内容を、画面に落とし込む

情報整理

ご提供資料や既存のエクセルファイルを参考にして、デモ画面を作成します。
実際のPJでは、**貴社ご担当者様と議論を重ねながら、画面内の優先順位や重要なポイントを整理**していきます。



この画面において、弊社が大事だと感じたポイント

- ユーザーの属性によっては、欲しい情報が異なると思われる
- 全ての情報を1画面に収めると、可読性が落ちて、欲しい情報を受け取りにくいのではないか
- 検索機能により、案件を素早く見つけたいニーズがありそう
- 誰がいつ・何の作業を行うかを明確にし、進行状況を把握しやすい
- 作業の追加や必要工数の確認・管理を簡単にする
- 他社に依頼している工程も含め、案件全体を可視化したいのではないか
- 各作業のステータスを明示し、進行の問題を早期に把握できること
- スケジュールの調整を柔軟に対応可能なインターフェースであること

ユーザーの理解

ご提供資料と合わせて、ユーザーにとって本当に使いやすいか、ヒアリングを通じて整理し、UI/UXに落とし込みます。
そのため、弊社では簡易ペルソナを作成し、ユーザーの解像度を高めた状態で画面を作成します。

Persona

ペルソナ（生産管理）



名前

高崎 公平

年齢

40歳

性別

男性

収入

600万円

移住地

神奈川県

勤務先

製造業 生産管理

在籍期間

16年

内向的

外向的

分析的

直感的

せっかち

おっとり

リテラシー低

リテラシー高

ITリテラシー

・エクセルの基本的な操作は可能。

・ショートカットやコピペにも可能。

・難しい操作や高度なITリテラシーはない

スケジュールの見たいポイント

・ 中期(3ヶ月～1年)

・ 量産立ち上げスケジュール

・ 設備導入計画

・ 短期(週～月単位)

・ 具体的な生産計画

・ 試作スケジュール、部品の納期

仕事内容

・ 生産計画をエクセルで管理している。

・ 進捗を確認し、納期を調整している。

・ 在庫を管理し、必要に応じて発注している。

・ 設計・調達・製造のスケジュールを全体調整。

・ OEM・サプライヤーとの調整を行い、部品の遅延や生産能力不足を防ぐ。

・ 工場のキャパシティやリソースを見て、生産負荷を調整。

現在の環境・状態

・ 設計や製造のスケジュールに変更がある場合一つずつ修正対応している。

・ 生産管理の担当者は、納期管理やコスト削減に追われ、資材遅延や人手不足に対応しながら業務を進める。

・ デジタル化が進む一方、現場調整やデータ活用が課題となり、柔軟な対応と効率化が求められている。

求めていること

・ 顧客からの指示通りの注文を滞りなく完遂したい

・ 複数業務のスケジュールをわかりやすく確認したい

・ スムーズな動線・操作で簡単に管理を行いたい

Persona

ペルソナ（現場）



名前

古谷 幸太郎

年齢

30歳

性別

男性

収入

500万円

移住地

千葉県

勤務先

製造業 現場

在籍期間

7年

内向的

外向的

分析的

直感的

せっかち

おっとり

リテラシー低

リテラシー高

ITリテラシー

・エクセルの基本的な操作は可能。

・ショートカットやコピペにも可能。

・パソコンには慣れている。

期間

・ 短期(週～月単位)

・ 具体的な生産計画

・ 試作スケジュール、部品の納期

・ 超短期(日～シフト単位)

・ 量産立ち上げスケジュール

・ 設備導入計画

仕事内容

・ 生産作業：製品の組立、加工、検査、梱包などを行う。

・ 機械・設備の操作：生産設備を操作し、製造工程を進める。

・ 品質チェック：製品の寸法や外観を確認し、不良品を見分ける。

・ 材料・部品の供給：必要な材料や部品を準備し、工程に供給する。

・ 生産ラインの調整：トラブル発生時の対応や、工程の微調整を行う。

現在の環境・状態

・ 製造業の現場では、人手不足や納期の厳守、品質管理のプレッシャーが増大。

・ 自動化やデジタル化が進む一方、新技術の習得や現場対応の負担も大きい。

・ 効率化が求められ、柔軟な対応が必要な環境となっている。

求めていること

・ 担当作業に関わるスケジュールをわかりやすいこと

・ 具体的な作業スケジュールを想起しやすいこと

・ スムーズな動線・操作で作業の完了登録などを行いたい

求めていること

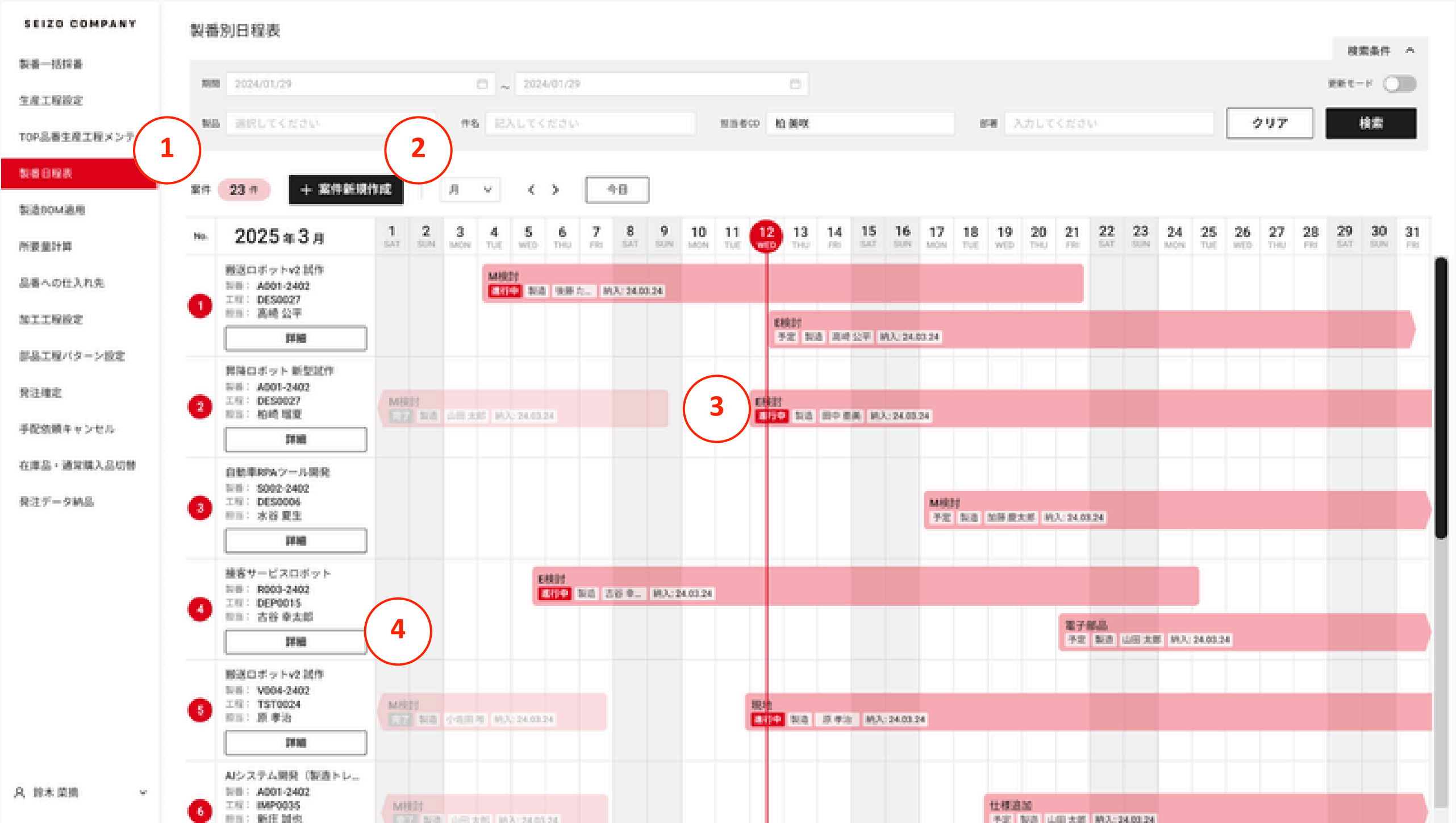
- ・顧客からの指示通りの注文を滞りなく完遂したい
- ・複数業務のスケジュールをわかりやすく確認したい
- ・スムーズな動線・操作で簡単に管理を行いたい

求めていること

- ・担当作業に関わるスケジュールをわかりやすいこと
- ・具体的な作業スケジュールを想起しやすいこと
- ・スムーズな動線・操作で作業の完了登録などを行いたい

日程表一覧（サンプル）

スケジュールの全体像を素早く把握し、進行管理の情報を直感的にインプットしやすいを心がけて作成します。

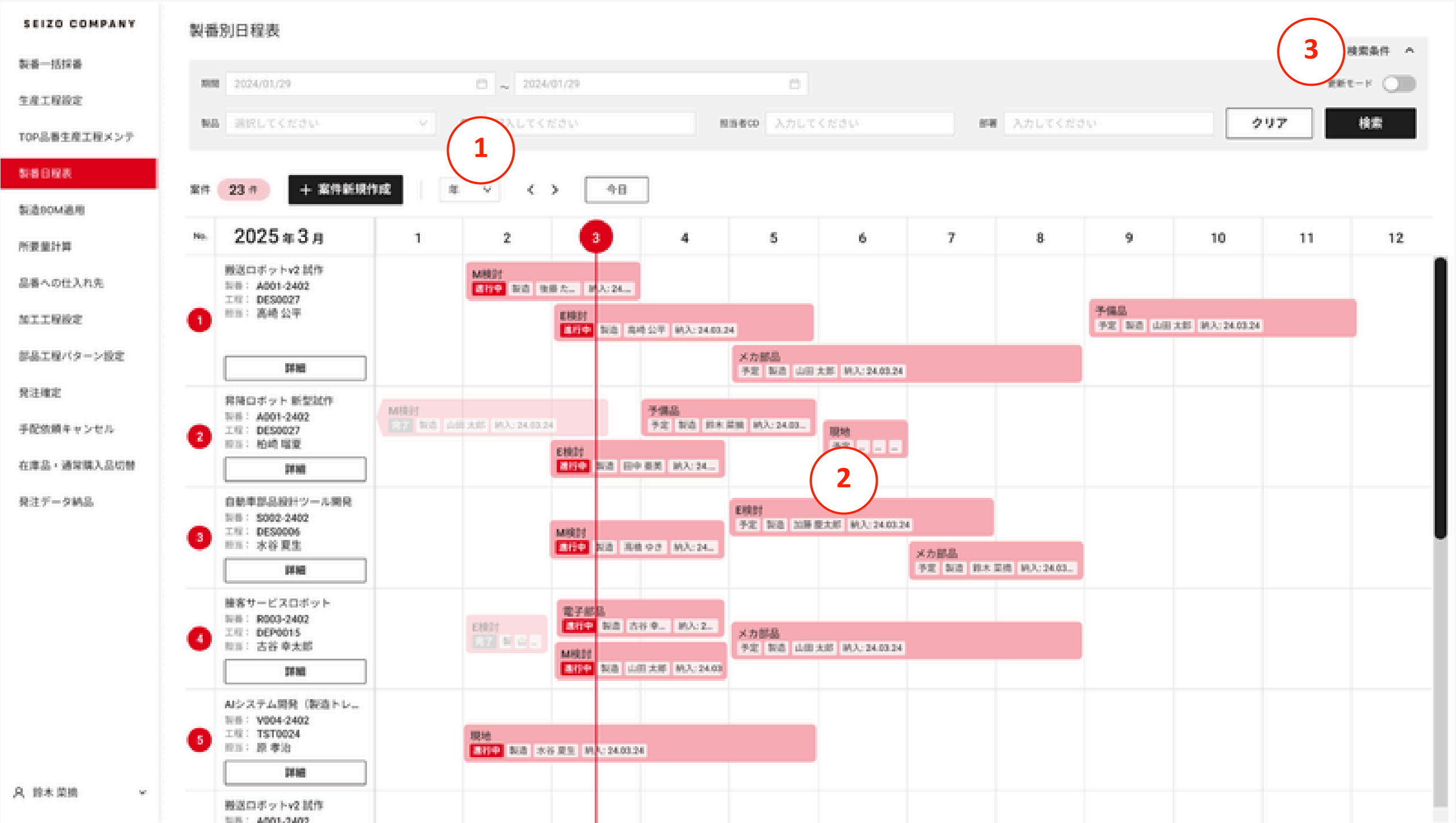


画面デザインのポイント

- 1 スケジュールの全体像を直感的に把握できるレイアウトを採用し、スムーズな確認をサポート。
- 2 案件のスケジュール追加ボタンを目立つ位置に配置し、操作の分かりやすさを強化。
- 3 必要な情報を最適な配置で整理し、進行管理のインプットがスムーズに行えるようにする。
- 4 ボタンやアクション要素の視認性を向上させ、操作ミスを最小限に抑制。

日程表一覧（サンプル）

生産管理者向けに、スケジュール全体を把握しやすい一覧と機能を提供いたします。

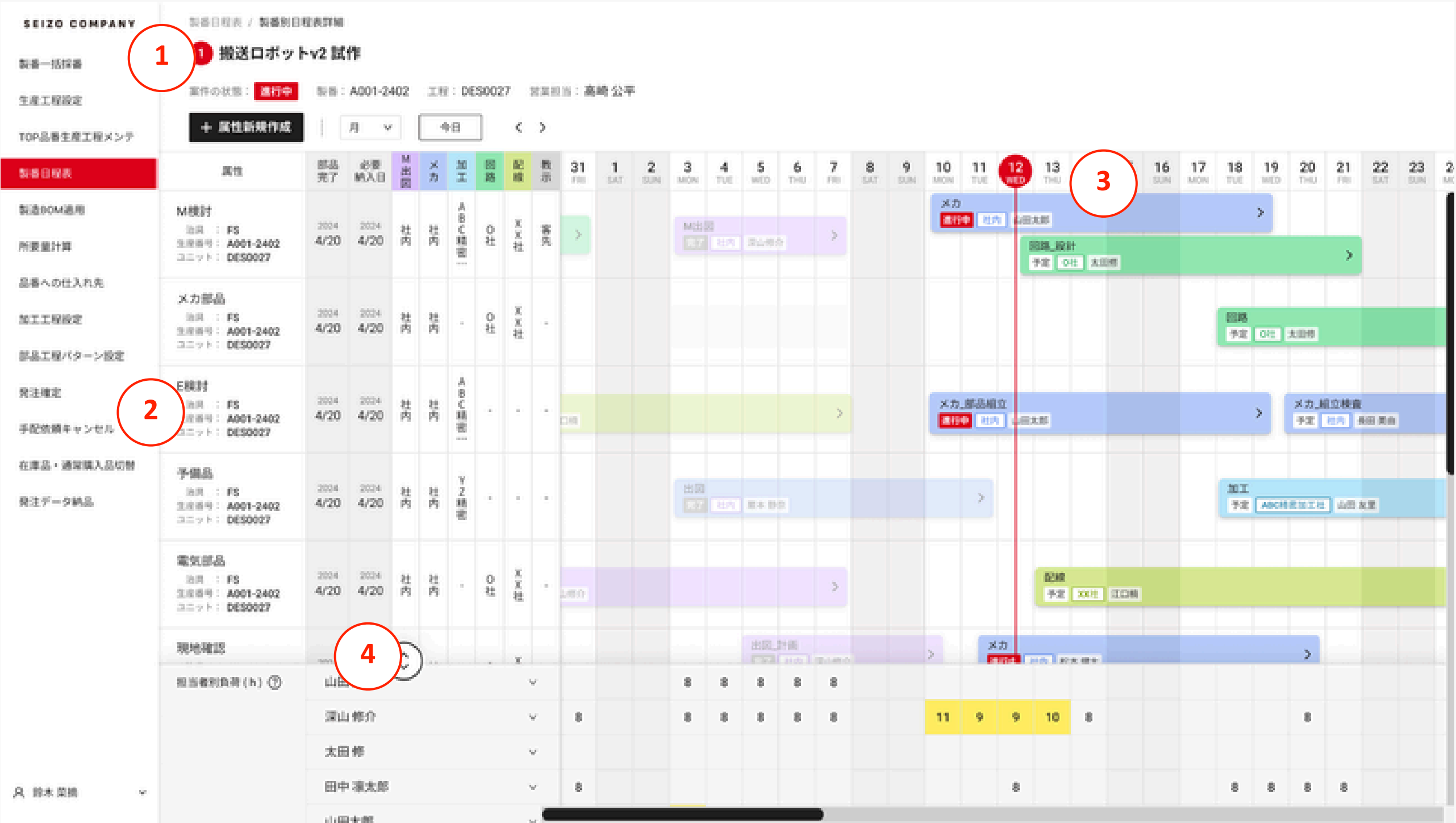


画面デザインのポイント

- ① 週・月・年単位の表示切り替えにより、短期・長期の計画を柔軟に管理。
- ② 年単位の表示に切り替えると、矢羽（アイコンや進捗表示）が小さくなるため、全体のスケジュール感を維持しつつ詳細を確認可能。
- ③ 画面上部に検索条件を設け、特定の案件を素早く見つけれられるよう設計。

日程表詳細（サンプル）

詳細画面では、現場の人が自身のスケジュールを把握しやすいように作成いたします。



画面デザインのポイント

- 1 選択した案件の詳細スケジュールを確認でき、No.と紐付けて覚えやすく整理。
- 2 「誰が」「いつまでに」工程を遂行するのかが把握でき、作業の前後関係や依存関係も視覚的に示し、全体の進行状況を簡単に理解できる。
- 3 工程を色分けし、社内外のスケジュールを直感的に識別できる。
- 4 担当者ごとの負荷は画面下で確認でき、より多くの人数を把握したい場合は、ドラッグ&ドロップで画面範囲を広げることができる。

日程表詳細（サンプル）

工数の調整を日付と紐付けて確認ができるように作成いたします。



画面デザインのポイント

- 1 担当者ごとの負荷やスケジュール調整をスムーズに行い、効率的な進捗管理をサポート。
- 2 プロジェクト全体のスケジュール調整や優先順位を明確にし、作業の円滑な進行を促進。

顧客課題と弊社ソリューション



課題 | パッケージシステムをカスタマイズして利用していたが、ビジネス環境の変化に柔軟に適應できない。

解決 | ユーザー中心設計のスクラッチ開発で生産管理システムを再構築することで、ステークホルダー・経営・システム部門・現場にとって最適なシステムを導入。



課題 | 「物流の2024年問題。」によりトラックドライバーの時間外労働に上限が設けられた。また非常にタイトなスケジュールのため、緻密なシステム構築方針策定能力と、スピード感を持った開発が必要。

解決 | 庫内作業負荷とドライバーの待機時間の削減を両立するスケジュール管理システムを導入し、トラックドライバーの業務効率化を実現。Sun*日本とベトナムの合同チームで、アジャイル開発の手法を用い、短納期且つ、ビジネス要件を押さえたシステム構築を実現。



課題 | 新規事業として「運送会社向けの配車支援サービス」を構築したいが超上流の支援から開発構築までを対応するリソースが不足していた。

解決 | Sun*のBTC（Business、Technology、Creative）機能を最大限に利用することで、サービスの立ち上がりからリリースまでの伴走支援。



課題 | トラック配送の予約/受付（バース管理）の開発を計画していたが、内製だけではスピード感の確保が困難。

解決 | コンサル機能と開発機能を持っているSun*の伴走支援。

産業用装置の設計・製造を行う
大手企業の製造部門

課題 | 全社の製造部門のDX計画策定にあたり、各地の工場の現状把握（AsIs）をする必要があるが、社内リソースが不足している。

解決 | Sun*UI/UXのデザイナーが、オンラインとオフラインで、現場をヒアリングし、業務フロー図を作成

Case Studies

導入事例：4社の支援実績

世界一のりん青銅メーカーを目指す企業の挑戦：りん青銅が紡ぐデジタル革命

事業の中枢を担う生産管理システムのスクラッチ開発支援

1952年創業の原田伸銅所は、進化を続ける社会を「りん青銅」で支える中堅企業です。りん青銅は、人間の身体に例えるなら血管のような役割を果たす金属であり、スマートフォン、PC、電子機器やICカードなど、IT分野では欠かせない素材です。近年、5G通信を基盤とするIoTやEV車、自動運転の普及に伴い、需要が拡大しています。これまで製造管理の中枢を担っていたシステムは、既存のパッケージシステムをカスタマイズして対応していましたが、需要の拡大に伴い、生産の効率化と働き方改革を実現するためにシステムの刷新を決断。スクラッチによる開発を上流～下流まで、Sun*の日本とベトナムのチームが一気通貫で支援いたしました。

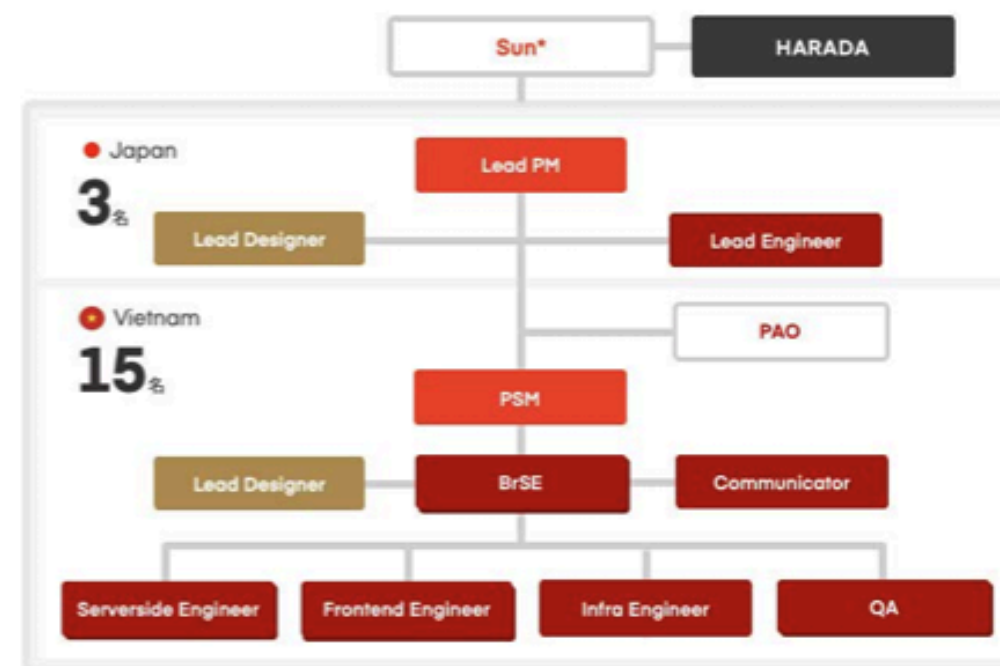
基本設計 開発実装 アジャイル開発 AWS DevOps UI/UXデザイン SMB 製造

ISSUE クライアントの課題

既存のパッケージシステムは、自社独自の受注生産型の業務フローに合わず、精緻な管理ができず、属人化しブラックボックス化していた。また、改修に時間がかかる課題も抱えていた。これらの課題に対して、生産性を高め、技術ノウハウを蓄積し、自動化が図れる独自のシステムを新規に構築したい。

SOLUTION 課題に対するSun*の対応

複雑で専門性の高い業務を理解するため、日本人メンバー全員で現地工場を訪問し、業務内容をしっかり把握したうえでシステムを設計。同時にデザイナーが要件をヒアリングし、モック画面をデザイン。ベトナムの開発チームも用語集を作成し、業務理解を深めつつ、機能の提案なども積極的に行った。



クライアントの声

『本気で課題に挑む人たちと、事業を通して社会にポジティブなアップデートを仕掛けていくこと』をミッションとするSun*さんは、企業の課題に対して真摯に耳を傾け、高い理解力と分析力、自らの都合に依らない適切な提案力を遺憾なく発揮してくれました。そのおかげで、難解な局面も無事に乗り切ることができました。また、開発期間中のビジネス変化にも迅速に対応いただき、期待以上のシステムが完成しました。ミッションを体現する人たちとの出会いに感謝しています。



物流の2024年問題、ドライバーの待機時間や労働環境を改善するDX

PULL型物流の実現に向けたWMS・TMSを機能補完するシステム化支援

トラックドライバーや運送業界にも働き方改革が適用されることで、物が運べなくなる可能性が生じるとされる「物流の2024年問題」。2024年4月からトラックドライバーの時間外労働の960時間上限規制と改正改善基準告示が適用されるなか、ダイキン工業社の配送センターにおいても、荷揃え待ちやバース開放待ちがセンターの業務都合でスケジュールが組まれていたため、無駄なドライバーの拘束時間が発生。これらを解消するために、庫内作業負荷とドライバー待機時間削減を両立するマスタースケジュールをつくり、庫内作業の進捗に合わせてドライバーを入場させるためのスケジュールを管理するシステムをSun*で開発支援いたしました。



- 基本設計
- 開発実装
- PPA
- アジャイル開発
- DevOps

- エンタープライズ
- 物流

Why? Sun* Sun*が選ばれた理由

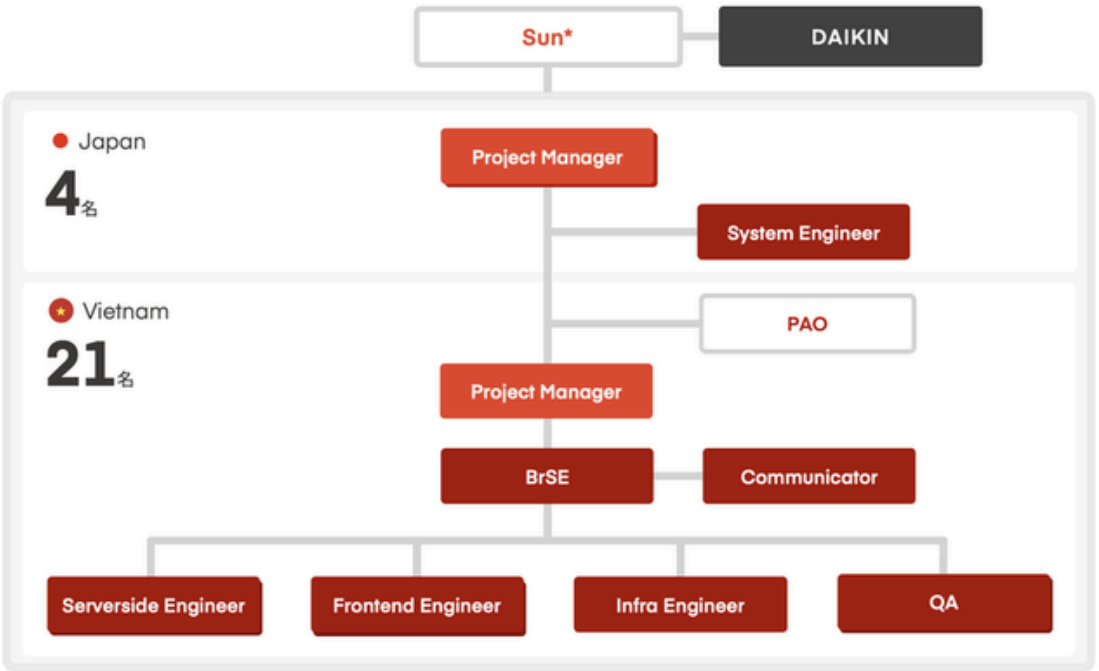
時間に余裕がなかったため、**アジャイル開発**を活用し、スピーディに設計から開発、業務定着化の支援の実績があった点。
また、**物流業界のシステム開発の経験のあるシステム設計者だけでなく、物流業務・オペレーションにも精通したメンバーがいたため**、解像度が高く、未来を見据えた提案を行えたこと。

SOLUTION 課題に対するSun*の対応

基幹システムと連携してデータを授受する部分は通常開発工期がかかるため、暫定的にRPAツールを活用して機能を実現し、後のステップでAPIを連携。また、クラウドアーキテクチャに対するセキュリティ要件に対して、情報システム部門と密に連携しながらシステムを構築していった。

クライアントの声

『Sun*をパートナーとして選んだ理由は、同じ組織にデザイナーがいてUIに関する技術を持っていることと、今回の物流システムと近い開発実績をお持ちだったこと、そしてスピードが感じられた点です。限られたスケジュールの中で、早期にキャッチアップし、プロジェクトを進行いただきました。また、予期せぬ開発にも迅速に対応していただき、弊社のビジネスに適した改善提案をしていただき、パートナーとして非常に信頼しています。』



ソフトバンク（株）と日本通運（株）の合併会社であるMeeTruck社の新規事業開発支援

配車業務をペーパーレス化し、 基幹配送業務のDXに向けた挑戦

MeeTruck社の創業事業となる「運送会社向け配車支援サービス」の立ち上げをSun*のUXチームが包括的に支援。ユーザーインサイトの抽出、サービス・ブランドデザイン、MVP開発、機動力が高い本開発推進など、プロジェクトの初期段階から事業開発に伴走。一体感あるチームで短期間でのサービスインを実現。



事業共創

エンタープライズDX

ブランド設計

ユーザー中心設計

アジャイル

USER INSIGHT

BRAND DESIGN



ユーザーインサイトを抽出し、
ブランドデザイン・サービスデザインへ反映

PROTOTYPING/MVP



プロトタイピング・MVP開発を通して早期
にユーザーからのFeedbackを獲得

APP DEVELOPMENT



開発チームのスケラビリティを活かし、
柔軟でスピーディな本開発

PASSIONATE TEAM

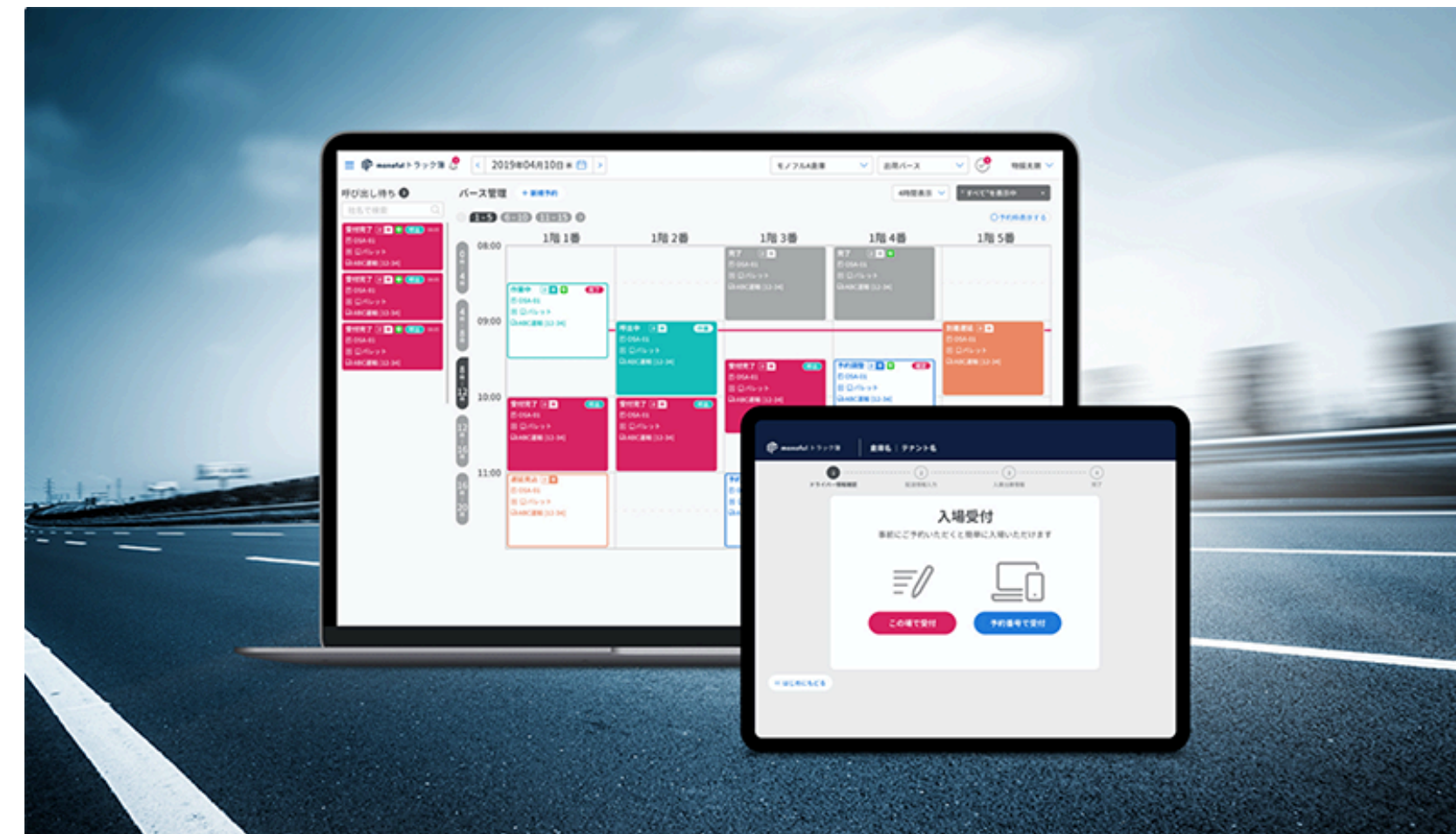


熱量あふれる顧客との一体感あるチーム

トラック配送の予約／受付（バース管理）サービス「トラック簿」の開発支援

物流業界を変革する 新規事業を半年で立ち上げ

世界有数の物流プラットフォームカンパニー、GLP(Global Logistic Properties)社のソフトウェア開発会社として立ち上がった株式会社モノフル。物流施設を利用するトラックの稼働効率を向上させる事を目的とした新規サービスの立ち上げを、事業企画からサービスの設計、開発までをクライアントとともに伴走。



共創型事業開発

ワークショップ

UI/UXデザイン

コンセプトメイキング

アプリ開発

Client Voice

「内製していたらこのスピード感は無理なので、そのフレキシビリティ・スケーラビリティはすごいなと思います。相談がくるかと思ったらいきなり形になっていたり、すごく助かりました。コンサル会社や大手SIerの要素を兼ね備えた開発会社って、なかなかないと思います。」

CTO SUPPORT



CTOの代理サポート。技術リードだけでなく、ビジネス面における企画からエンジニア採用までを支援

APP DEVELOPMENT



日本拠点とベトナム拠点で最適な開発チームを組成し、クライアントとともに、実装を伴走

RECRUITMENT SUPPORT



クライアントの自社内でも開発、運用を推進できるように、エンジニア組織の構築を採用面で支援

AIのアプローチ

レガシー刷新をAIで半自動化し、安全な移行を支援する「Regen*」

レガシーが課す税 — 見えないコストの4象限

セキュリティの脆弱性

フレームワークがEOLを迎え、脆弱性のバックログが蓄積します。年々システムのリスク露出が増加する一方で、適用できるパッチは減少し続けます。

エンジニアの採用難

古いスタックを知るエンジニアが減少し、採用に時間がかかります。システムの知識が少数のメンバーに集中し、強い属人化と退職リスクを生み出します。

ビジネスチャンスの損失

システムの硬直化により、新機能のリリース速度が著しく低下します。過去のアーキテクチャ設計が、現代のトラフィックパターンや要件に対応できなくなります。

維持コストの複利

システム変更のコストが雪だるま式に高くなります。レガシーインフラの維持そのものに開発予算が割かれ、本来のプロダクトロードマップが進行できなくなります。

何もしないことが、**最も高い支出**になる

なぜリニューアルは失敗するのか



挙動の未文書化

ビジネスルールは誰の頭の中にも明文化されていません。暗黙知に依存したまま開発が進むことで、重大な仕様の抜け漏れが発生します。



数%の特殊挙動の見落とし

誰も説明できないわずかな数%の特殊な挙動こそが、実は売上や顧客体験に最も影響を与える核心部分であることが多いのです。



コアロジックの欠落

新システムが旧システムの挙動を完全に再現できない。その瞬間、顧客は離れ、プロジェクトは失敗に終わってしまいます。



リニューアルの失敗は
「コーディング」の問題ではない。

「意味の保存」の
失敗である。

真のコスト方程式 — 現状維持 vs 従来型リライト vs Regen*

現状維持 (5年)



直接コスト

保守費の増大 + 障害対応



セキュリティ露出

年々累積しリスク悪化



ロールバック性

該当なし (劣化の一途)

従来型リライト



直接コスト

当初予算の2～3倍が典型



失敗リスク

ビッグバン移行で高リスク



ロールバック性

ビッグバン = 戻れない

Regen*



直接コスト

予測可能・段階的投資



失敗リスク

各ステップにゲートを設け
リスクを最小化



ロールバック性

モジュール単位で
完全可逆なプロセス

総支出は同じだとしても、**リスク**は大きく異なります。「階段を登る」確実なリニューアルへ

AIが変えた3つのこと（2026）

2026年現在、これまで自動化が困難だった3つのボトルネックが解消されました。



1. 仕様の復元

従来は数週間かけて書いていたドキュメントを、ソースコードから高精度で抽出。暗黙知を構造化し、「正解パターン」を復元します。



2. コードの翻訳

単なるシンタックスの変換ではなく、アーキテクチャやイディオムを理解した上でモダンスタックへの変換を行います。



3. パリティテスト

復元された仕様からテストを自動生成。新旧のシステム両方で実行し、挙動の同等性を完全に担保する強力なゲートとなります。

リニューアルの経済原理は反転した。「精度」と「速度」の両立へ

AIツールだけでは不十分 — 規律が差を生む



オートコンプリート中心

コーディングの速度は上がっても、リニューアルにおいて最も重要な「全体を説明する事前のプラン」を描くことはできません。



文脈保持の限界

単独のAIツールでは、プロダクト全体のコンテキストが分散してしまうため、正しい挙動のコード生成が難しくなります。



監査証跡とパリティの欠落

生のツールには、旧システムの挙動に対する同等性（パリティ）の意識や、全変更に対する厳格な監査ログを残す仕組みがありません。



AIツールを導入するだけで
解決するほど甘くない

ギャップは「知性」ではなく
「エンジニアリングの規律」
にある

3フェーズのプロセス — 各ステップに「ゲート」を設ける

意思決定を容易にし、どの段階でも成果物を保持して停止可能なリスク最小化のアプローチ。



各フェーズに受け入れゲート。どのフェーズの後でも、成果物を手にして止める選択が可能です

フェーズ0 — 仕様を凍結する

コードを1行書く前に、レガシーシステムから「正解パターン」をドキュメントとして復元します。

docs / specs /

// レイヤー 1: アーキテクチャ文書 (9ファイル)

-  system-overview.md
-  route-list.md
-  data-model.md
-  screen-list.md
-  ... (他5ファイル)
- 

// レイヤー 2: 機能ごとの詳細スペック

 F001_UserAuthentication/

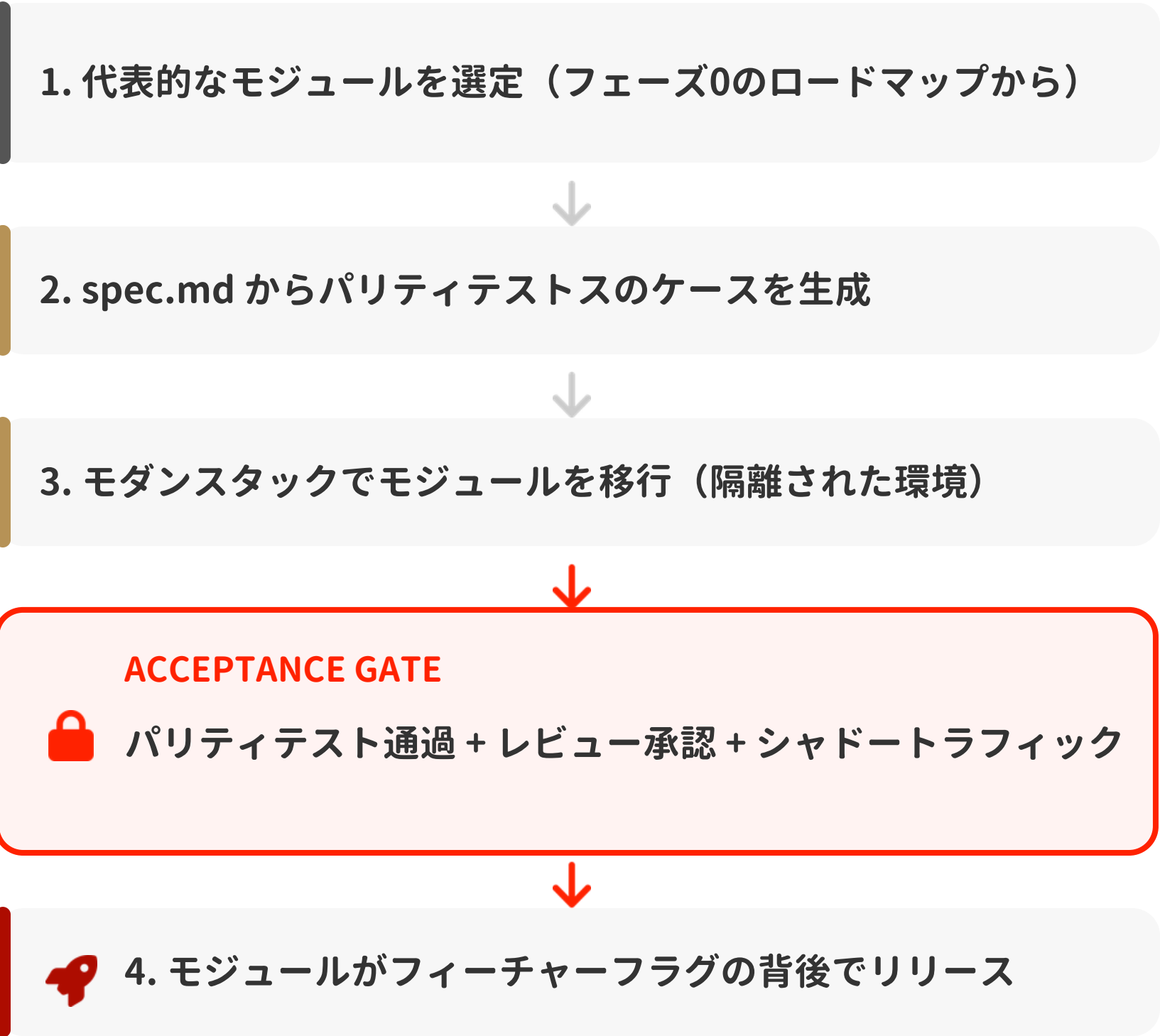
ドキュメント優先のアプローチ

実装を開始する前に、すべての仕様をMarkdown形式で明文化します。これにより、チーム全体で「何を作るべきか」の共通認識を確立します。

- 不確実な仕様を早期に発見・排除
- レビュー可能な「設計図」の作成
- AIコード生成のコンテキストとして活用

フェーズ1ー 速度とパリティを証明する

パイロットモジュールの選定から、パリティ（同等性）の証明を伴う厳格なリリースまでのフロー。



妥協のない受け入れゲート

モジュールは次の条件すべてを満たした場合にのみ本番環境へリリースされます：

- パリティテストによる**同等性の担保を確認**
- シニアレビューでの**P0（致命的欠陥）発見ゼロ**
- 7日間のシャドートラフィック差分が許容範囲内

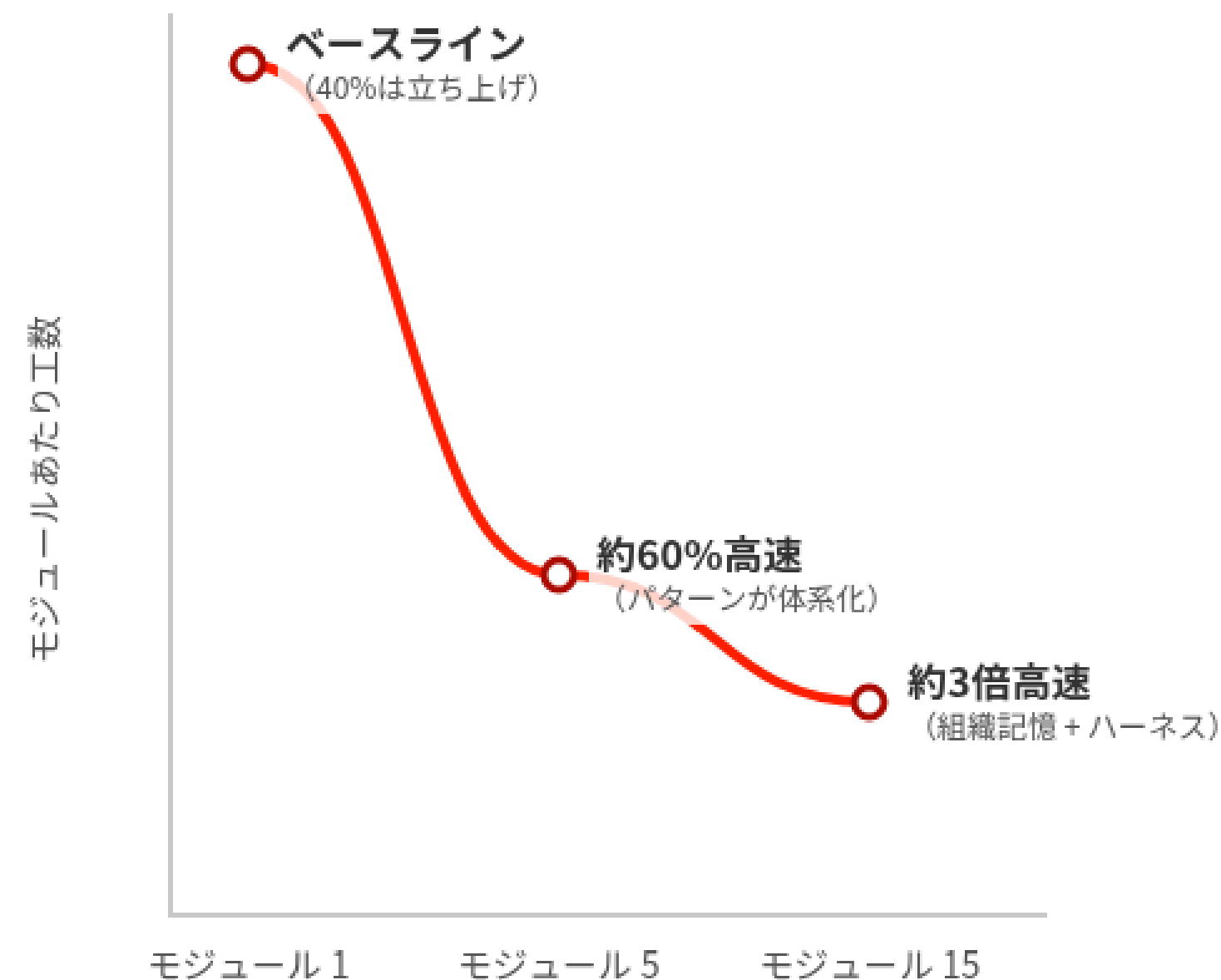
成果へのコミット

ゲートを通過するために必要な超過工数は、Sun*によって負担します。

パリティテストは新コード記述「前」にフェーズ0のスペックから生成されるため、受け入れ基準が客観的であり、スコープへのコミットが可能です。

フェーズ 2 – 複利の速度

案件が長くなるほど、エンジニアリング知見が蓄積され、モジュール移行の速度は複利的に向上します。



速度が複利的に向上する理由

プロジェクト固有のイディオムを保持する
エンジニアリングメモリ（記憶）

モジュール間で使い回せる
再利用可能なパリティハーネス

各モジュール完了時の振り返りによる
学習の固定化（組織記憶の定着）

4か月目以降の複数モジュール並列実行

マイルストーンゲート付き T&M

各ゲート（パリティ通過 / レビュー / デプロイ）の達成に対して対価が発生。

プロジェクト規模が大きかつ長くなるほど、プロジェクト全体のコスト効率が高くなります

高い互換性の提供 — パリティなしにリリースされない

スペックからテストを生成し、旧システムと新システムの両方で実行して完全に一致させる厳格な規律。



5つの規律（エンジニアリング・ガバナンス）

テストはスペックから生成 — 新コードからではなく、元のシステムからのテスト生成。

まず旧システムでテストを実行し、ベースライン挙動を確立。推測ではなく測定。

新モジュールはベースラインを再現してからマージ。挙動ドリフトは許可しない。

シニアレビューを強制 — Sun*シニアSEがP0発見に対する拒否権を持つ。

シャドートラフィックで検証 — 本番カットオーバー前に実トラフィック負荷で確認。

“

同一性を担保し、
ビジネスを守る

エンジン紹介 — Takumi

ここまでご覧いただいたプロセスは、Sun*独自のAI Agentエンジン「Takumi」で動いています。



17の専門エージェント

プランナー、実装者、テスター、シニアレビュアーなど、各領域に特化したエージェントが連携し、役割分担を持ってプロセスを推進します。



55以上のスキル

スペックリバース、パリティテスト生成、コード翻訳など、Sun*の刷新ノウハウを具体的なスキルとして実装し、高い精度と速度を実現します。



メモリレイヤー

単発のセッションで終わらず、プロジェクト全体のコンテキストや意思決定の履歴を保持し、長期にわたる一貫性と再現性を担保します。

単なるAIチャットではない。規律を自動化し、品質を再現する「エンジニアリング・ハーネス」

得意とするスタックと移行プレイブック

各レガシースタックに対して実証済みの「移行プレイブック」を用意し、ハマりやすい落とし穴を事前に回避します。

レガシースタック	特有の課題	移行プレイブックの焦点	モダンスタック
PHP CodeIgniter, 古いLaravel	Fat Controllerの蔓延 フロントエンドとの密結合 N+1などのパフォーマンス問題	レイヤードアーキテクチャの導入 Eloquent ORMへの安全なデータ移行 APIとフロントエンドの明確な分離	Laravel (最新) React / Vue
Java Struts 1 / 2, 古いSpring	EOLフレームワークの脆弱性 密結合なJSPによる変更困難 重厚なXML設定のメンテナンス	Spring Bootへのシームレスな移行 アノテーションベース設定への変換 JSPからREST APIへの段階的リライト	Spring Boot Next.js / Nuxt
.NET ASP.NET WebForms	ViewState肥大化による遅延 Windows基盤への強い依存 ユニットテストの実施困難	.NET Coreでのマルチプラットフォーム化 MVC / Web APIパターンへの再構築 クラウドネイティブ設計への刷新	.NET 8 Azure / AWS
Ruby 初期 Rails (3 / 4)	暗黙のコールバック地獄 gemの依存関係の陳腐化 Fat Modelの限界と保守難	ServiceObjectによるロジック分離 陳腐化したgemのモダン代替への置換 RSpecテストスイートの自動生成	Rails 7 Hotwire / SPA

 上記以外のスタックや独自フレームワークについても、アセスメントフェーズにて柔軟な移行戦略を策定可能です。

他社アプローチとの比較

大手SIの「規律」とAIツールの「速度」を統合。監査証跡とパリティ保証を伴う、次世代の刷新プロセス。

評価軸	大手SI	オフショア	純粋AIツール	Regen*
速度・スピード	⚠ 低 重厚で硬直化したプロセス	— 中 人海戦術によるカバー	✓ 高 コーディング単体は高速	★ 大幅に高速 AI自動化による複利成長
柔軟性 (スコープ変更・可逆性)	✕ なし 仕様凍結後の変更困難	— 中 コミュニケーションコスト増	✓ 高 都度のプロンプト変更は容易	★ 完全可逆 ゲート制による完全可逆進行
エンジニアリング規律 (品質・コンテキスト保持)	✓ 高 厳格だが属人的な管理	⚠ 低 品質のバラつきが大きい	✕ なし プロジェクト全体の記憶欠如	★ 体系化された規律 16のAIEージェント+シニアSE
パリティ / 監査証跡 (同等性の確実な保証)	— 中 手動テスト中心で漏れリスク	✕ 低 100%の互換性担保は困難	✕ なし 新旧の比較機能を持たない	★ 保証あり 自動テスト生成 + 厳密なログ

結論：速度と柔軟性、そして厳格な規律。これら全てを統合したのが「Regen*」の独自のアプローチです。

契約の全体像とスケジュール

最短3週間でアセスメントを完了し、4ヶ月目には継続的なリリースサイクルへ。
各フェーズ末に意思決定ポイント（ゲート）を設けています。



一か八かのビッグバン移行からの脱却



各フェーズは独立しており、お客様は各段階の結果を見た上で、次のステップに進むか、あるいはそこで止めるかを安全に判断いただけます。

まずは4週間の「フェーズ0」アセスメントから

御社のシステムを完全に可視化し、リスクのない安全な刷新への第一歩を踏み出します。

期間



3～4週間からスタート

フェーズ0の納品物

- ✓ **二層スペックライブラリ**
Markdown + PDF形式でシステムの仕様を完全文書化
- ✓ **技術的負債およびセキュリティ登録簿**
- ✓ **モダナイゼーション・ロードマップ**
モジュールの分解と最適な移行順序の策定
- ✓ **フェーズ1および2の見積**

NEXT ACTION

- 1 対象システムの情報を共有する**
本日中に、システムの概要と優先的に刷新したいモジュールをご共有ください。
- 2 48時間以内にプランを受け取る**
具体的なアセスメント計画とSOW（作業範囲記述書）のドラフトをご提出いたします。

※ 万が一フェーズ1へ進めない場合でも、社内で数ヶ月分の価値に相当するドキュメントをお渡しいたします。

つくるだけでは、終わらせない。 使われるシステムへ、UXとAIで。

Sun*は、現場のUXリサーチとAI駆動のレガシー刷新を組み合わせ、
実際に手を動かしながら"使われる"業務システムへの刷新を支援する開発パートナーです。

[Sun*に相談する](#)