

アシストソリューション研究会中日本 2022年度分科会発表会

# ＼DXに乗り遅れるな！／ ノーローコード開発ガイドの 策定とその有用性

ノーコード／ローコード開発と適用方針 ～適用方針に関する研究～ 分科会  
チーム・ノコロコ（青木、森山、田中、武田、小久保、宮澤、松尾）



# 目次

## I. 分科会紹介と研究概要

- 分科会紹介
- 研究概要
- 活動実績

## II. ノーコード/ローコードツールとは

- ノーコード/ローコードとは何か
- ノーコード/ローコードツール比較

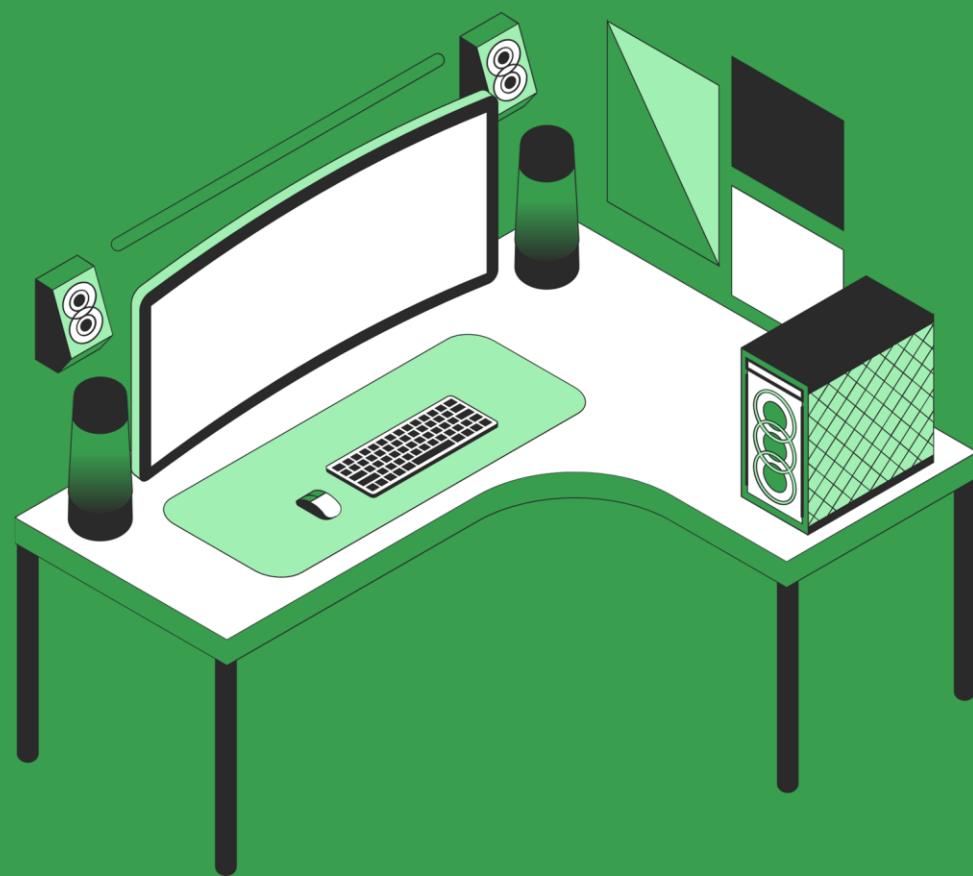
## III. ノーコード/ローコード開発

### スタートガイドと有用性

- スタートガイドの策定
- 作成したスタートガイドの有用性検証
- 研究結果

## IV. 総括

- 総括
- 展望



# I. 分科会紹介と研究概要

# 分科会紹介



# 知識・経験が多様なチーム“ノコロコ”メンバー

🏆 リーダー



リコージャパン株式会社  
**青木 優太**

🏆 サブ  
リーダー



トランコムITS株式会社  
**森山 欽司**



鈴与システムテクノロジー株式会社  
**武田 雅美**



愛知県経済農業  
協同組合連合会  
**田中 佑治**



株式会社メイテツコム  
**小久保 響子**



株式会社アシスト  
**宮澤 祐平**

事務局  
窓口



株式会社アシスト  
**松尾 成昭**

# 知識・経験が多様なチーム“ノコロコ”メンバー

🏆 リーダー



楽々FW使ってます

🏆 サブ  
リーダー



高速開発ツール  
色々使ってます



楽々FW・WF、  
ASTERIA Warp使ってます



PMメインで活動中  
ノーローコードは  
未経験



スクラッチ開発メイン  
ノーローコードは  
未経験



運用管理ツール担当  
ノーローコードは  
未経験

事務局  
窓口



前職はPG,今はPM  
ノーローコードは  
未経験

## まずはミッションで想いを1つに

ノーコード/ローコード開発の現状と、各メンバーの知見やこの分科会に対する想いを掛け合わせて、当分科会のミッションを以下としました。

“ 「本分科会で研究した内容をガイドラインとしてアウトプットし、私達と同じようにノーコード/ローコードツール活用に悩める人達に貢献したい！」



# 研究概要





# 私たちの研究テーマ

## 事務局からのテーマ

開発  
／  
運用

### ノーコード／ローコード 開発と適用方針

～適用方針に関する研究～

東  
中  
西

コーディングが不要あるいは最小限のコーディングでアプリケーションを開発する「ノーコード／ローコード開発」が日本でも使われ始めています。ノーコード／ローコードが必要とされる背景を探るとともに、その有用性や特性・適用分野などを調査し、野放し状態で開発が進まないよう、適用方針案を策定することで本格的なノーコード／ローコード開発時代の到来に備えます。

- ノーコード／ローコードが必要とされる背景
- ノーコード／ローコード開発が適用できる／できないシステム
- ノーコード／ローコード開発の適用方針の策定

## 私たちの研究テーマ

「ノーコード／ローコードツールのスタートガイド策定」  
ツール選定だけでなく開発・運用も含めた勘所をスタートガイドとして  
まとめてリリースし、その有用性を検証する



# ノーコード/ローコード開発の理解を深めると見えてくる問題点

まずはノーコード/ローコード開発についての理解を深めました。

市場動向の把握  
ツールの理解

業務課題を洗い出し  
対象業務課題を決定

ノーコード/ローコード  
ツール比較

するといくつかの問題点が・・・

業務課題にどのツールを  
選定すべきかわからない

導入が難しい、実用に  
コーディングが必要など  
初心者には壁

ツールが使えるようになると  
属人化や野放しの  
問題が発生する

「野放し」とは・・・

- 組織内のツールの乱立
- 開発したアプリの乱立
- 開発後のブラックボックス化



# 解決すべき課題と研究テーマの決定

抽出した  
問題点

業務課題にどのツールを  
選定すべきかわからない

導入が難しい、実用に  
コーディングが必要など  
初心者には壁

ツールが使えるようになる  
と属人化や野放しの  
問題が発生する

解決  
すべき  
課題

選定の判断材料が必要

開発の知見が必要

運用ノウハウが必要

これらをふまえ研究テーマを決定

「ノーコード/ローコードツールのスタートガイド策定」  
ツール選定だけでなく開発・運用も含めた勘所をスタートガイドとして  
まとめてリリースし、その有用性を検証する



# 活動実績



# 定例会とチーム別活動

21 回の定例会 +

多くの個別活動

- 活動全体の検討、役割分担
- 進捗管理、課題管理

- リーダー、サブリーダー、事務局窓口で活動整理や方向付け
- チームごとにアプリ開発、スタートガイド作成、アンケート調査、発表準備

	活動日	活動内容	参加人数	参加率
発足会	2022/3/9	分科会発足会、メンバー顔合わせ	6名	86%
第1回定例会	2022/3/30	1. 各メンバーの業務課題ベスト3共有 2. ノーコード/ローコードツールの概念の調査結果共有 3. 研究テーマ、方向性検討 4. ワークショップ内容のフィードバック	7名	100%
第2回定例会	2022/4/8	1. ノーコード/ローコードツールの調査結果共有 2. 活動テーマ案・活動の進め方検討	7名	100%
リーダー会	2022/4/15	(活動整理、方向性・進め方協議)	3名	100%
第3回定例会	2022/4/22	1. ツールを触ってみた結果を共有 2. リーダー会の検討結果共有 3. 業務課題とツール検討結果共有 4. 活動テーマ案討議	6名	86%
第4回定例会	2022/5/13	1. 課題と対応ツール案共有 2. 活動テーマ案共有	7名	100%
リーダー会	2022/5/17	(活動整理、方向性・進め方協議)	3名	100%
第5回定例会	2022/5/27	1. 中間報告資料の内容レビュー	6名	86%
リーダー会	2022/5/30	(活動整理、方向性・進め方協議)	3名	100%
第6回定例会	2022/6/10	1. 中間報告会のフィードバック共有 2. ターゲットの検討 3. ガイドラインの対応範囲の検討	7名	100%
第7回定例会	2022/6/23	1. 「ユーザー部門」の定義共有・決定 2. ガイドラインの構成を共有・決定 3. 今後のスケジュール共有・調整	7名	100%
第8回定例会	2022/7/7	1. ガイドライン記載内容検討 2. ガイドライン作成 役割分担	6名	86%
第18回定例会	2022/12/9	スタートガイドブラッシュアップ版すりあわせ 成果報告書すりあわせ 最終発表会のシナリオ検討	6名	86%
第19回定例会	2022/12/19	スタートガイドブラッシュアップ版完成 成果報告書完成 最終発表会のシナリオ検討	6名	86%

定例会と個別活動、オンラインと対面を組み合わせるハイブリッドに活動



# 数字でみる成果

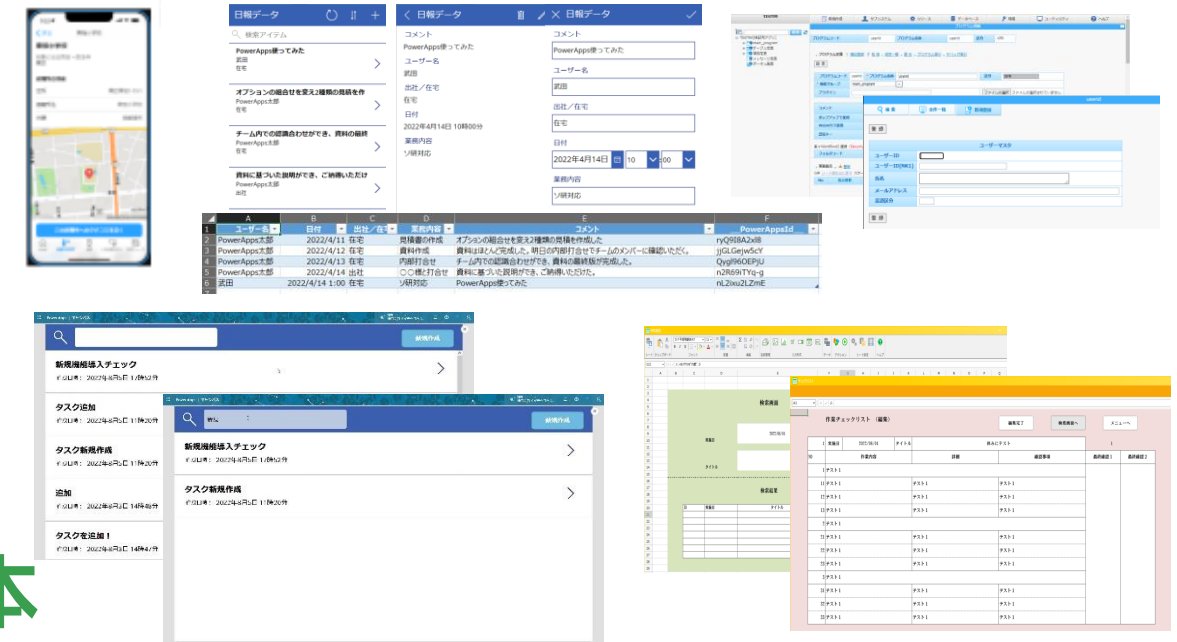
検証したツール

11種



開発したアプリ

11本



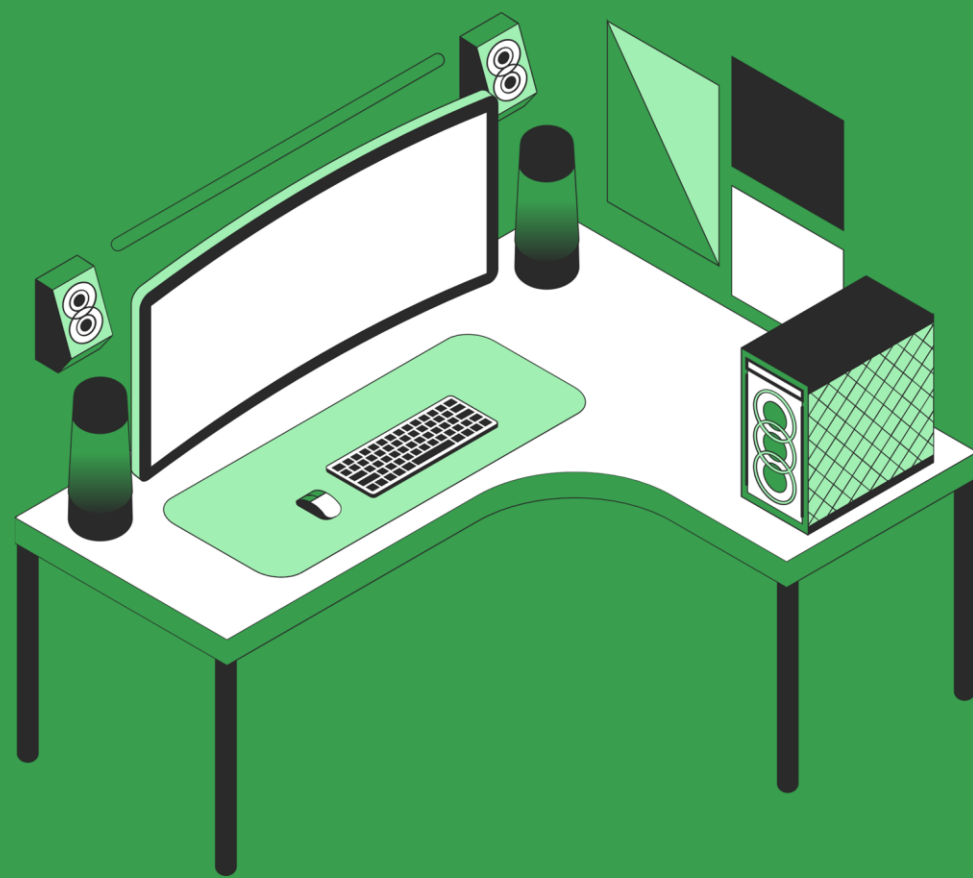
ノーコード/ローコード開発スタートガイド

84ページ



実際にツールに触れてアプリを開発するというリアリティと  
多くの人が使えらるものをアウトプットすることにこだわった





## II. ノーコード/ローコード ツールとは

ノーコード /  
ローコードとは何か





# ノーコード/ローコードの定義

市場動向  
ツール  
理解

対象業務  
決定

ツール  
比較

現在、各社が自社のツールを独自にノーコードかローコードに分類していますが・・・

誰でも使える！

ノーコードツール「XXX」

実際は、作りこみに  
プログラミングも必要・・・

本格的な開発ならコレ！

ローコードツール「XXX」

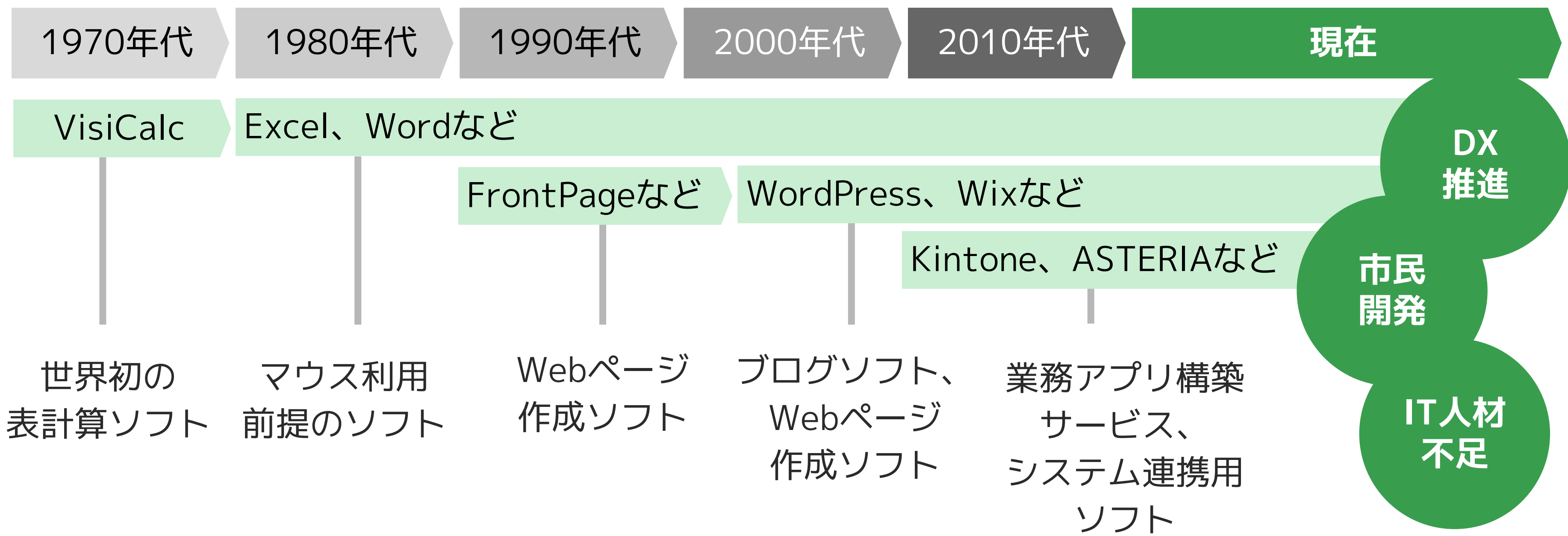
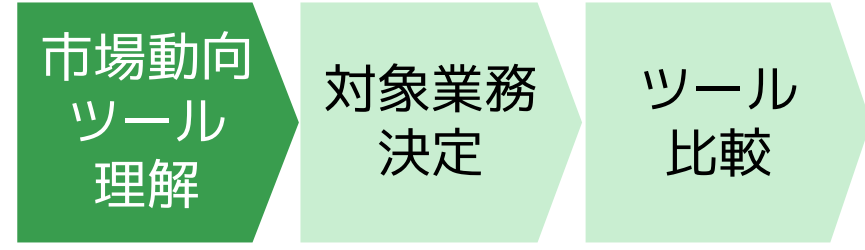
ローコードツールでも  
プログラミングしない使い方もある・・・

そこで私たちは、ツールの分類ではなく開発スタイルとして定義しました。

プログラミングの機能を利用しない開発：ノーコード開発  
プログラミングの機能を利用した開発：ローコード開発



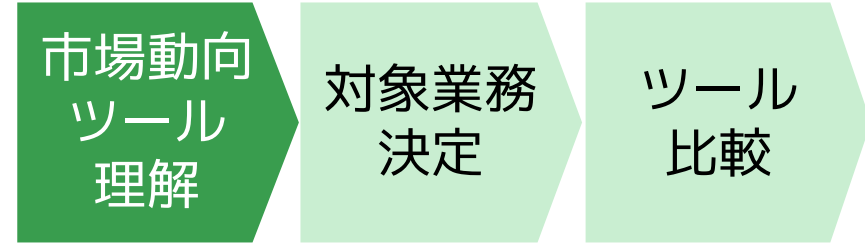
# 歴史と背景



半世紀前からコーディングを削減する努力がされていた  
現在、人手不足により非IT担当者による開発のニーズが高まっている

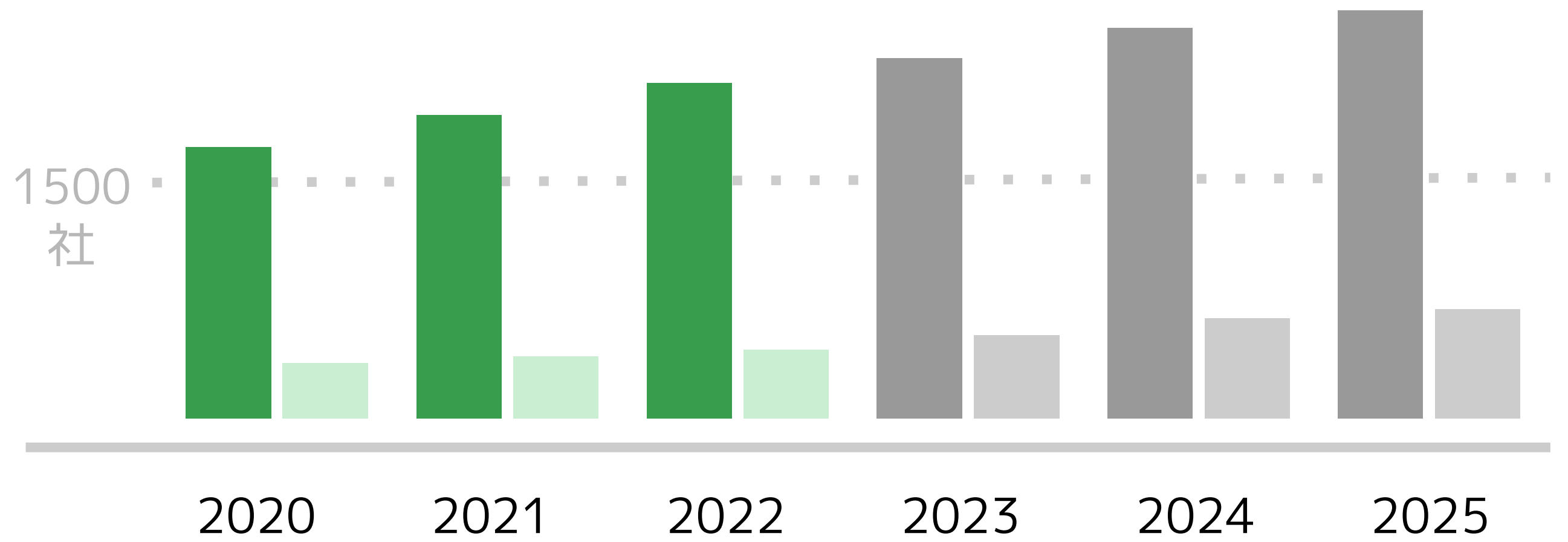


# ノーコード/ローコードツールと市場



ローコード開発ツール導入社数の実績と予測は以下の通りです。

- パッケージ型
- SaaS型

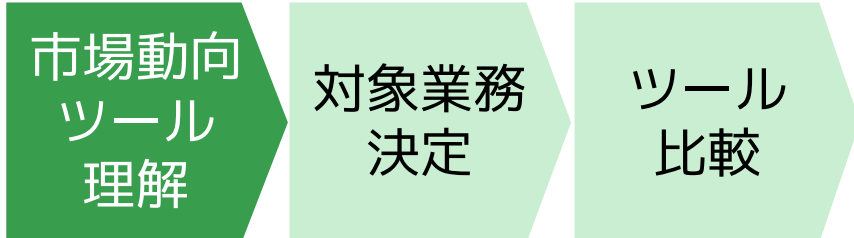


「富士キメラ総研 ソフトウェアビジネス新市場2021年度版ーローコード開発ツール」を参考に分科会にて作成

人手不足・短期開発・工数削減を背景に市場は右肩上がり  
昨今話題のキーワードとして認知度も高まっている



# まずはノーコード/ローコードツールを触ってみた



ノーコード/ローコード開発経験のないメンバーもいるため、まずは各自でノーコード/ローコードツールを触ってみました。

## ビジネスアプリ

## その他業務自動化

**Power Apps** (Microsoft) | 更新日: 2022/4/21 | 担当: 武田

**楽々Framework3** | 更新日: 2022/04/21 | 担当: 青木

ツール	強み	弱み
Power Apps	多様なデータソースとの連携、即ツールとの連携、Excelファイルでの管理よりも高度な管理申請業務	数百のデータソースに接続可能なアプリで、画面(UI)で開発が難しい
楽々Framework3	Java, CSS, JSで様々な業務・デザインに対応可能、親子関係のレコードをツリー形式で表示可能、テストデータの自動生成	非エンジニアによる開発、複雑な画面を短時間で開発、ミドルウェアなどの開発

8つのツール

**AppSheet** | 更新日: 2022/4/21 | 担当: 宮澤

**楽々Workflow II** | 更新日: 2022/4/21 | 担当: 田中

ツール	強み	弱み
AppSheet	マルチデバイス対応、データの表示、作成、更新、削除などシンプルなのは短時間で開発可能、各種Googleサービスとの連携が容易、導入料金は無料、RPAのような作業自動化機能もある(AppSheet Automation)	データを利用し、そのデータを元にアプリを作成する、カスタムな開発スタイル、GoogleスプレッドシートやSalesforce、AWS、Boxなど、あらゆるデータソースにアクセスしてアプリを開発
楽々Workflow II	電子申請、電子決済、データCSV出力、Excel連携(入出力項目)	フォントサイズの変更不可、経緯自由なイメージなのでやれないことは少ない(あまりわかっていないだけかもしれませんが)

**Asterio warp** | 更新日: 2022/4/21 | 担当: 武田

強み	弱み
利用プランにより異なる(番号が大きいほど上位) <ul style="list-style-type: none"> <li>Core: 簡易な業務自動化</li> <li>Standard: システム間のデータ連携・加工</li> <li>Enterprise: 大規模連携・大量データの取扱いの考慮</li> </ul>	利用プランにあまりないアダプターの利用、細かい設定が必要なアダプターの利用、複雑な画面や入力チェック

## データ連携

**bubble** | 更新日: 2022/4/21 | 担当: 武田

強み	弱み
ノーコードで、レスポンスが速いPC・スマホWebサイトを構築できる、アプリ内でデータベースの構築が可能、UI、処理の自由度が高いため業務向きのものも作成できる	ノーコードで、開発が難しい、複雑な画面を短時間で開発、ミドルウェアなどの開発

## モバイルアプリ

**Adalo** | 更新日: 2022/4/21 | 担当: 武田

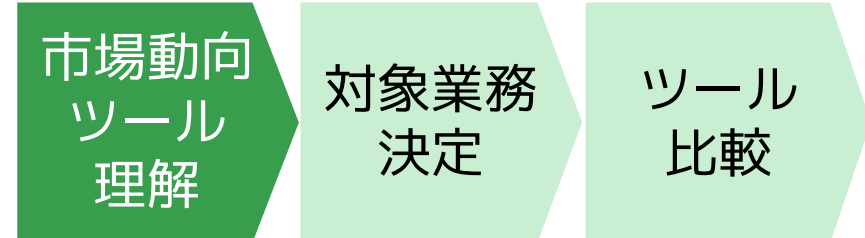
強み	弱み
データをためるデータベースの構築、オブジェクトの自由配置、多少の設計、一つのアクション(クリックなど)からできる	APIとの連携(調査すれば分かるかも)、もう少し深く掘り下げていないと出来ないのでは?という印象を持ってしまおう

**Glide** | 更新日: 2022/4/21 | 担当: 小久保

強み	弱み
データを投稿して収集する、地図上にピン打つようなマップ作成、ユーザ登録(ログイン)機能も作成可能	データの結合が指定できない、一覧も最後のページ(リレーション)以外には利用できない、配置の細かい調整等できない

## モバイルアプリ

# 例：Glideによる災害情報アプリの開発



ノーコード  
ローコード

## 【Glide】を触ってみた

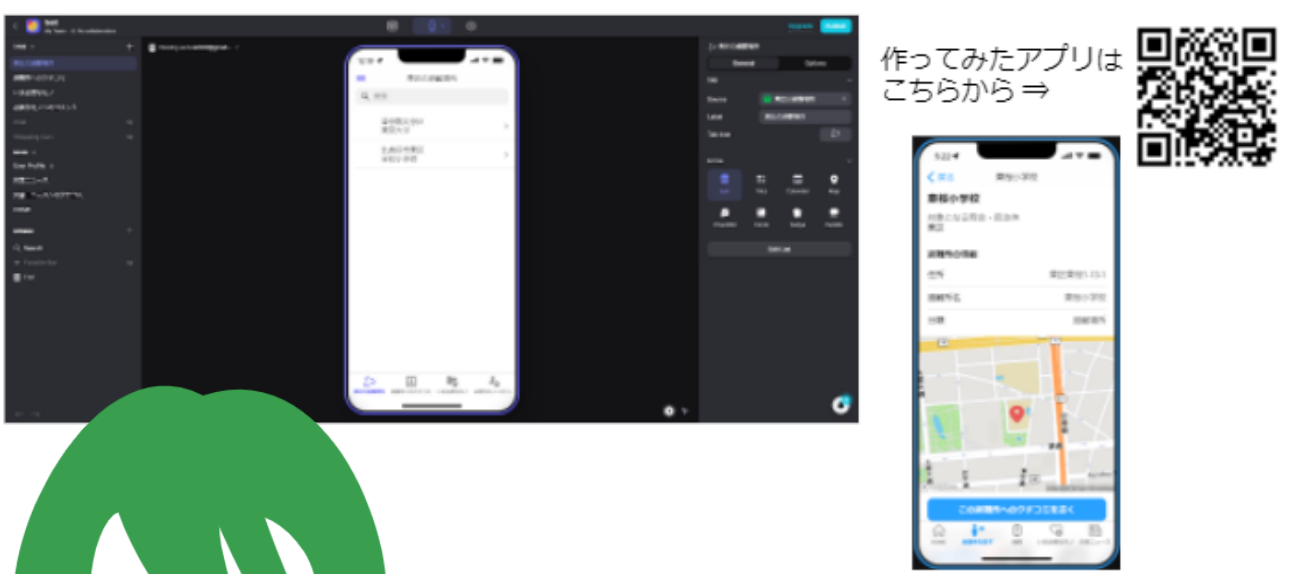


更新日：2022/4/21  
担当：小久保

基礎情報	
開発元 (国)	Glide
製品サイト	<a href="https://www.glideapps.com/">https://www.glideapps.com/</a>
提供形態・費用	SaaS ・ オンプレ <span>無償</span> ・ <span>有償</span>
機能・特徴	<ul style="list-style-type: none"><li>・ノーコードで、PC・スマホWebサイトを構築できる</li><li>・Googleスプレッドシート、Excelなどからアプリを作成できる</li><li>・業務向き、というよりはエンドユーザ向き(社内向けのライトなものなら十分な感じもする)</li></ul>

触ってみた評価	
やれること	<ul style="list-style-type: none"><li>・データを投稿して収集する</li><li>・地図上にピン打つようなマイマップ作成</li><li>・ユーザ登録(ログイン)機能も作成可能</li></ul>
やれないこと	<ul style="list-style-type: none"><li>・テキストの装飾が指定できない</li><li>・一覧も最低限のバリエーション以外は利用できない</li><li>・配置の細かな調整等はできない</li><li>・ユーザの種類による情報の出し分けはできない(かもしれない)</li></ul>
使い勝手	<ul style="list-style-type: none"><li>・すべて英語のため、翻訳を活用しながらの操作</li><li>・ハンズオンを一度やればコツは比較的つかみやすい</li></ul>
マニュアル、サポート、コミュニティなど	<ul style="list-style-type: none"><li>・公式マニュアル、コミュニティあり(残念ながらすべて英語でのやりとりになる)</li><li>・インターネット上の情報も多少はあり</li></ul>
想定される活用シーン	<ul style="list-style-type: none"><li>・例に作成したような情報を発信したり、収集するコミュニティサイトのスピード構築</li></ul>

### 画面イメージ/こんなアプリを作ってみた



#### その他自由コメント

- ・最低限の「こういう一覧作りたい！」がとても簡単に生成可能(学生時代に卒業研究で作成したアプリがこれだけで作れそう)
- ・ハンズオンがスマホアプリのみだったため、Webアプリも作成してみたい
- ・想像以上にスプレッドシートを読み込んだだけで「アプリだ...」という感想をもった

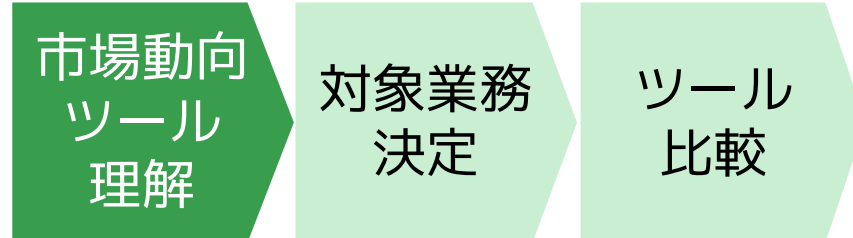
## アプリ動作イメージ



最新！いま避難所に必要なモノ(数字は緊急度)



# ノーコード/ローコードツールを触ってわかったこと



## 利点

SaaS型の場合、会員登録するだけで利用開始可能

多くのツールがドラッグ&ドロップで機能を実装可能

パラメータ設定やプログラミング  
・API連携で機能追加、拡張可能

掲示板や日記帳、SNSアプリ等は  
ノーコードですぐに構築可能

## 考慮事項

パッケージ型はインストールに時間がかかり、なかなか開発に辿り着かない

ドラッグ&ドロップは、実装数が多い場合に時間がかかることも

特定機能に特化したツールはカスタマイズが限られる

複雑な業務の実装、デザインの細かい装飾にはカスタマイズが必要

## 4つの観点

始めやすさ

操作性  
学習コスト

柔軟性

業務適合性

主に4つの観点で違いがあることがわかり、ツールを比較する際の軸に



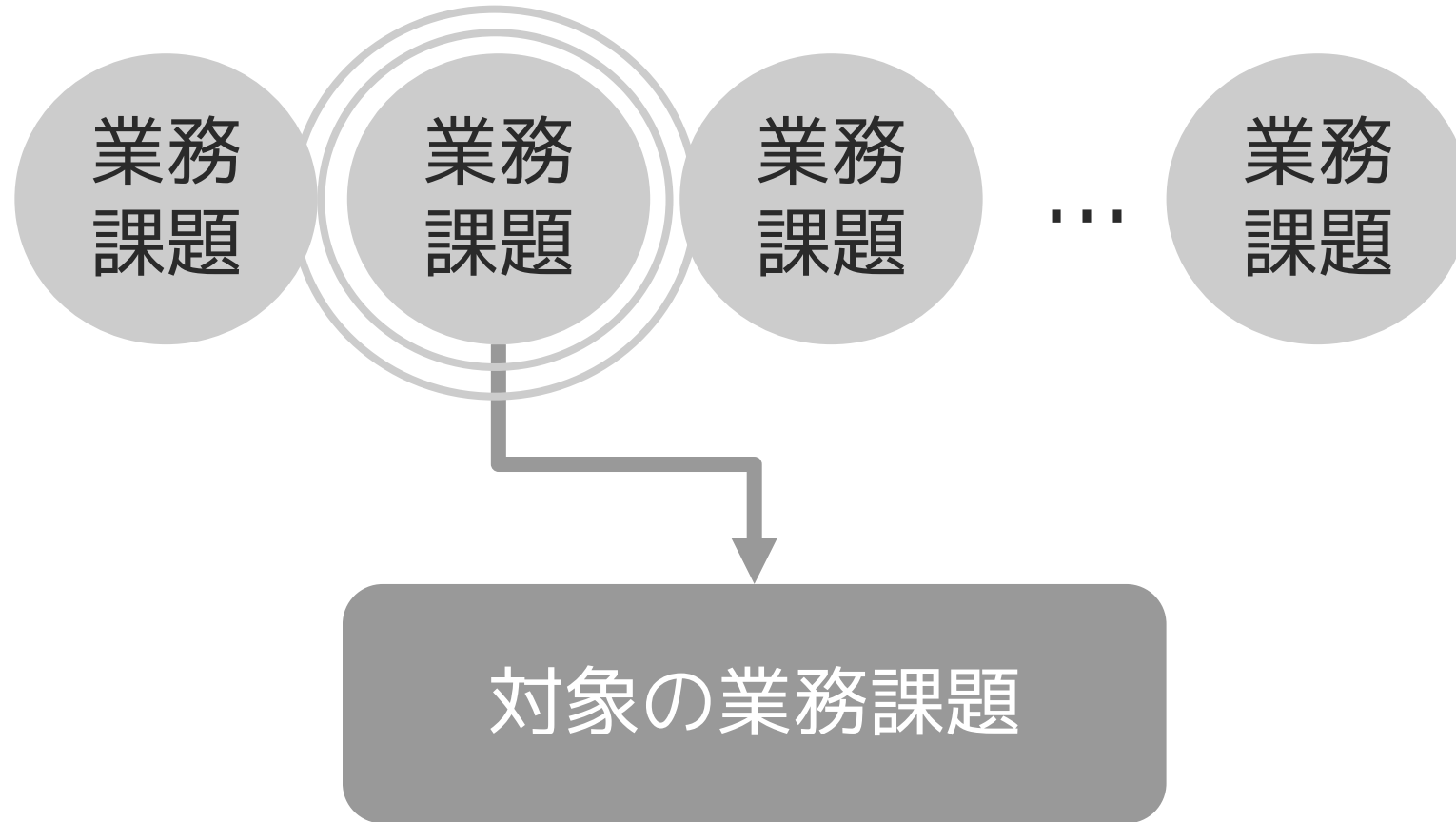
ノーコード /  
ローコード ツール  
比較



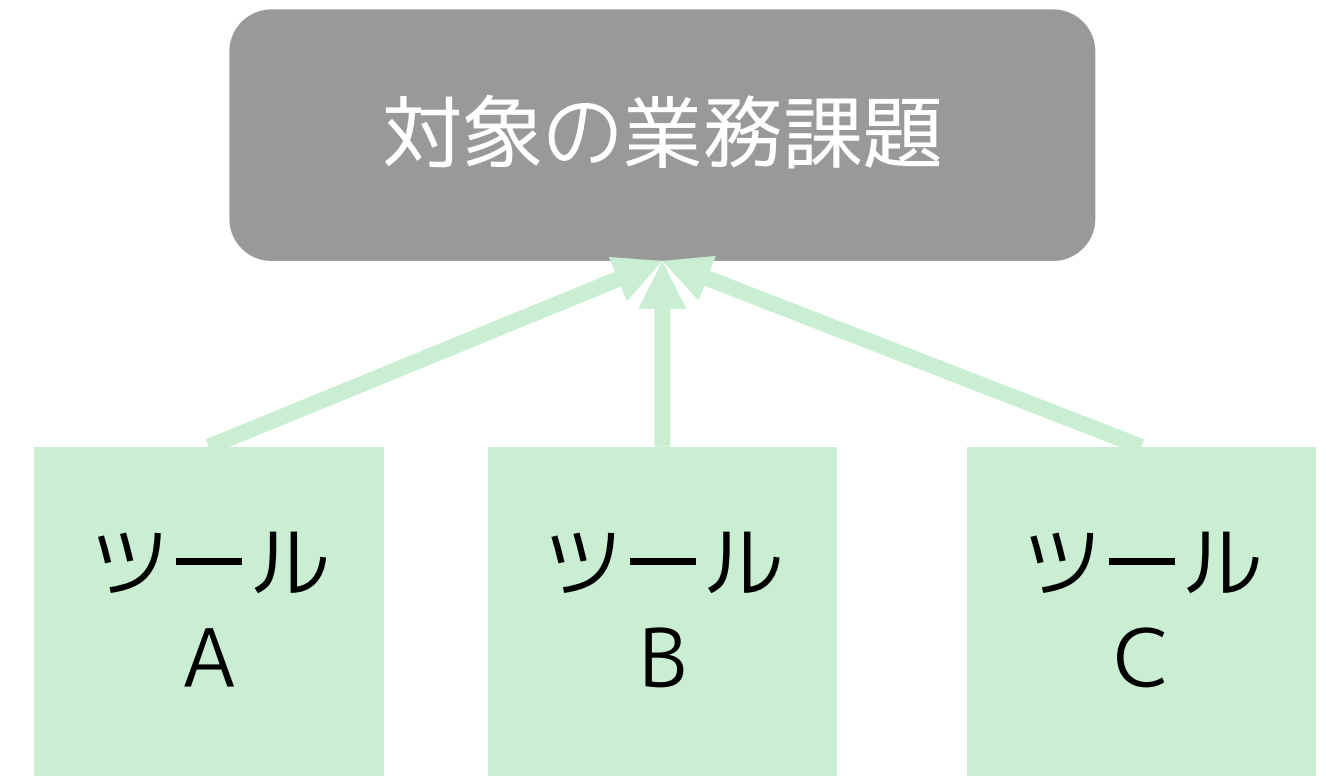
# 同一テーマにおけるノーコード/ローコードツール比較



自分たちが抱える業務課題から  
対象業務を設定



対象とした業務課題について  
複数のツールでアプリ開発



1つのテーマに対し、異なるツールでアプリを開発してみてその結果を比較





# チームメンバーが抱える業務課題の洗い出し

業務での困りごと・課題ベスト3を持ち寄ってみる

タスク管理に  
まつわる課題

スケジュールの5重管理

業務ルールが多岐に散在している

メンバーの業務の優先度、工数の管理

プロジェクトフォルダ内のファイル検索

ドキュメントと実装の乖離

オンライン打ち合わせ

開発案件の増加で対応しきれない

システム稼働通知メールのチェックが面倒

動画サイトへのアップロード管理

ペーパーレスにしたいがコストがかからない

意思決定から開始までの時間が長くて労力がとられる

メールやチャットで業務対応依頼があるが視覚的にわかりにくい

ノーコード、ローコードツールそのものにつわる課題

ビジネス課題系（ノーコード、ローコードで解決すべきもの）

ノーコードローコードの適用

ノーコード/ローコードツールの優位性、アピールポイント

ノーコードツールだといってもみんなが使ってくれない

ノーコードツールでは合わない業務は結局スクラッチ

ノーコードローコードの利用

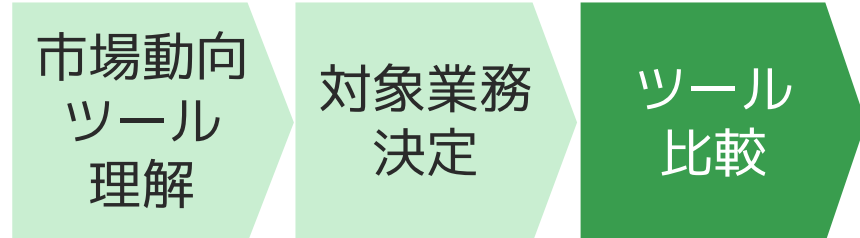
ノーコード/ローコード設計書の作成（フォーマット欲しい）

ツールを試したいが管理者権限を返してもらえない

タスク管理のためのチェックリストをアプリ化することを決定

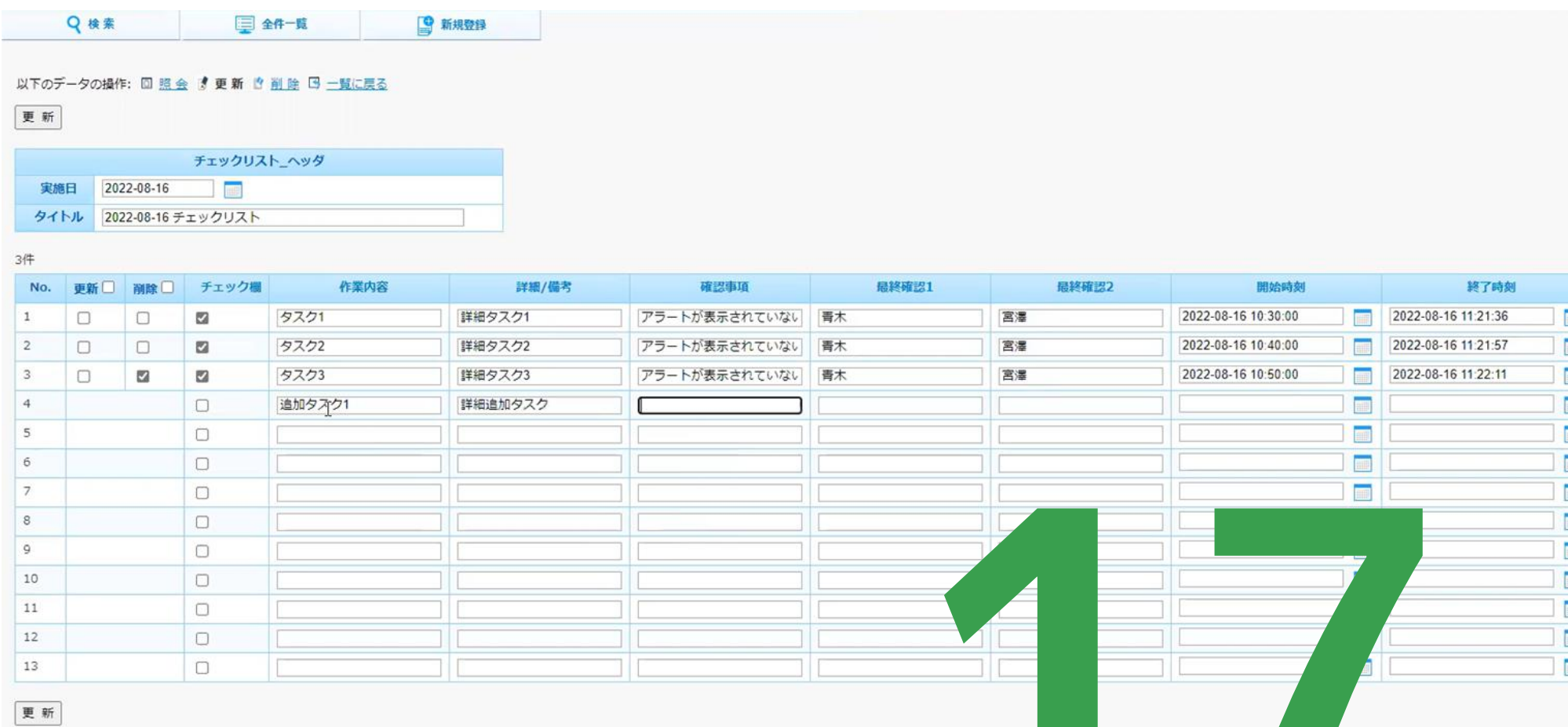
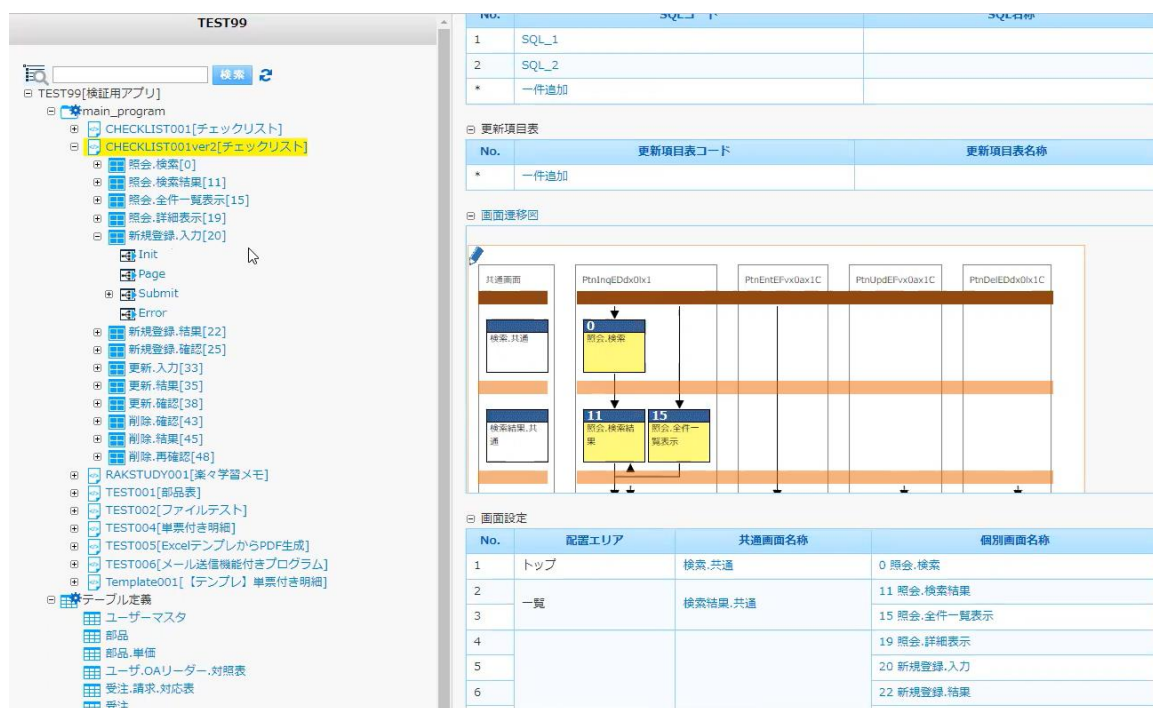


# ノーコード/ローコードツールでの開発結果（Aチーム）



## 楽々Framework3

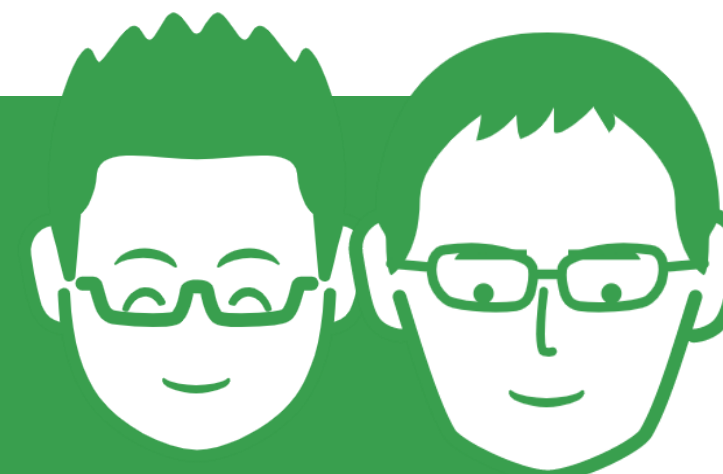
### 開発したアプリの画面



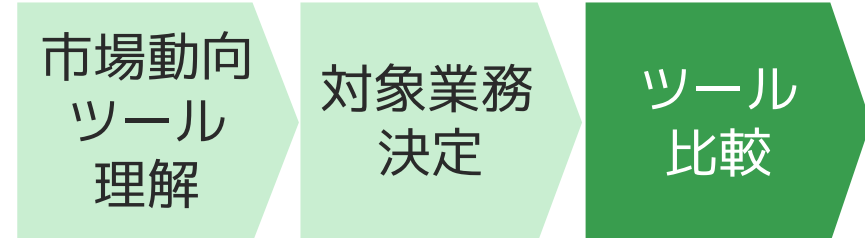
今回のテーマへの適合度：★☆☆☆☆

約 17 時間で開発

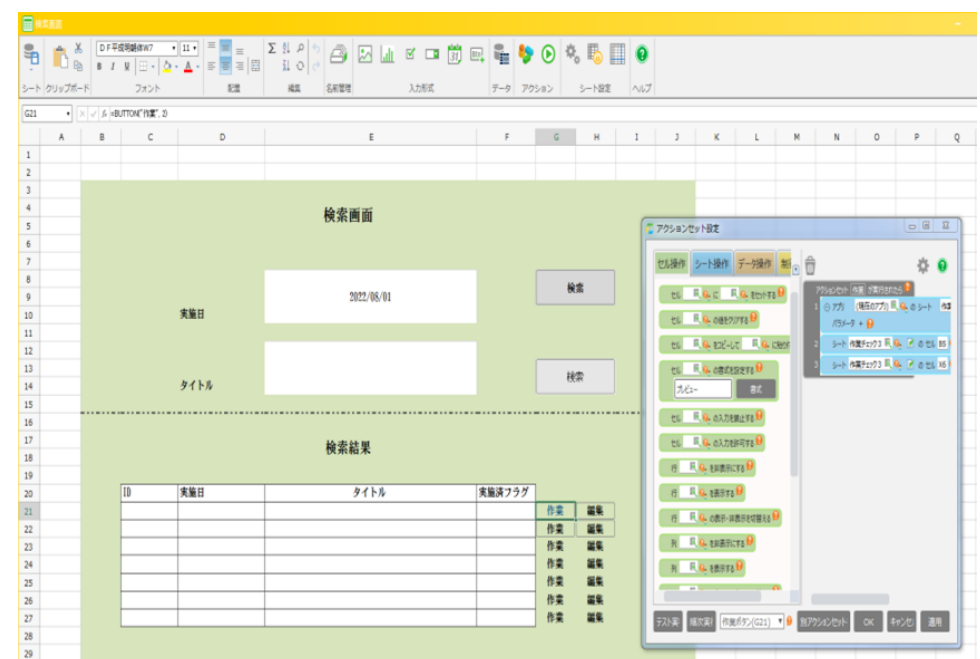
環境構築、カスタマイズに専門的な知識が必要  
複数画面に同じ設定を適用できるため、共通画面は作成しやすい  
学習、経済コストが高いため、少人数には不向き



# ノーコード/ローコードツールでの開発結果（Bチーム）



開発したアプリの画面



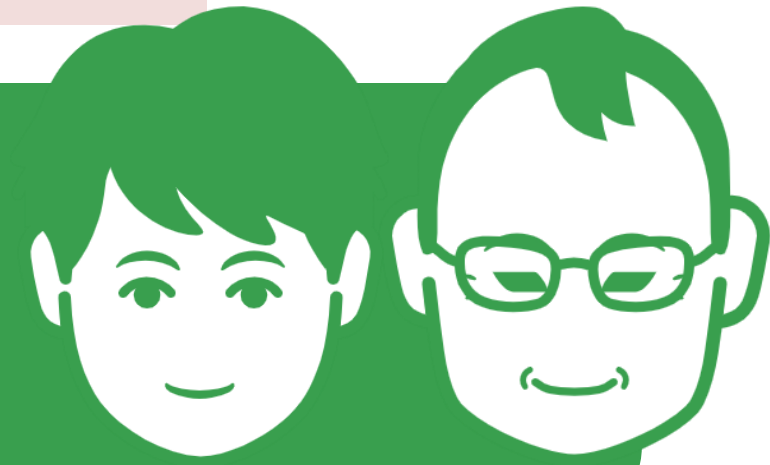
NO	実施日	タイトル	詳細	確認事項	最終確認1	最終確認2
1	2022/08/01	休みにテスト				
1	テスト1					
11	テスト1	テスト1		テスト1		
12	テスト1	テスト1		テスト1		
13	テスト1	テスト1		テスト1		
2	テスト1					
21	テスト1	テスト1		テスト1		
22	テスト1	テスト1		テスト1		
23	テスト1	テスト1		テスト1		
3	テスト1					
31	テスト1	テスト1		テスト1		
32	テスト1	テスト1		テスト1		
33	テスト1	テスト1		テスト1		

今回のテーマへの適合度：★★★★★

約 10

時間で開発

操作はExcelライクであり、また無料トライアルには操作説明動画が付属しているため、初心者でも始めやすい  
複雑な作りこみは難しいが個人や少人数の業務には向いている

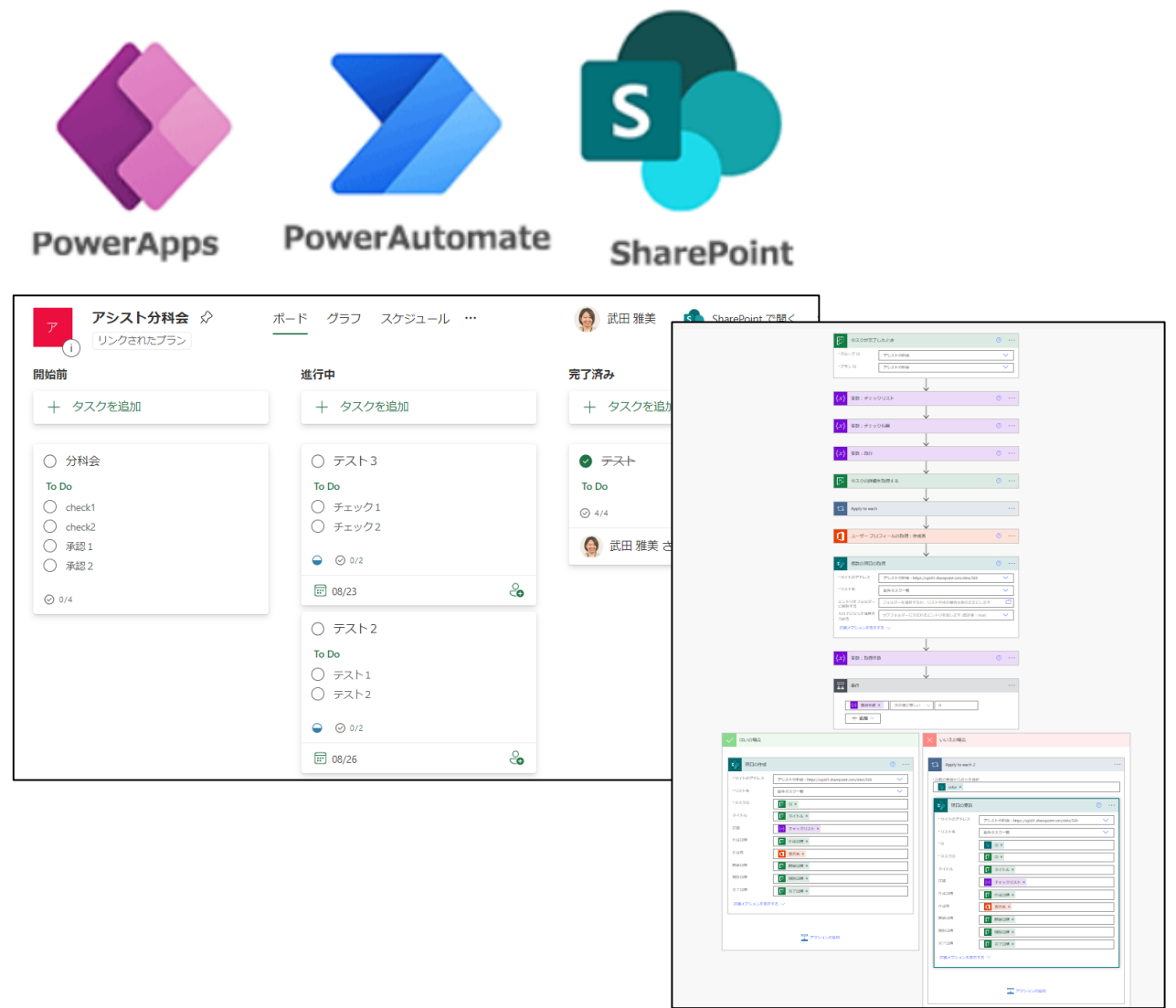


# ノーコード/ローコードツールでの開発結果（Cチーム）

市場動向  
ツール  
理解

対象業務  
決定

ツール  
比較



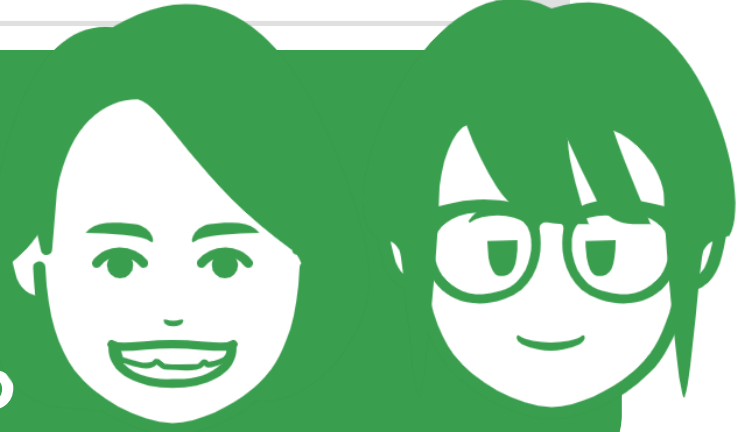
## 開発したアプリの画面



約 **23** 時間で開発

今回のテーマへの適合度：★★★★☆

サービス連携には一定の知識が必要、機能が多いため  
必要な情報を探すスキルも求められる  
ブラックボックス化の懸念もあるが少人数での利用に向いている



# ノーコード/ローコードツールの比較

市場動向  
ツール  
理解

対象業務  
決定

ツール  
比較

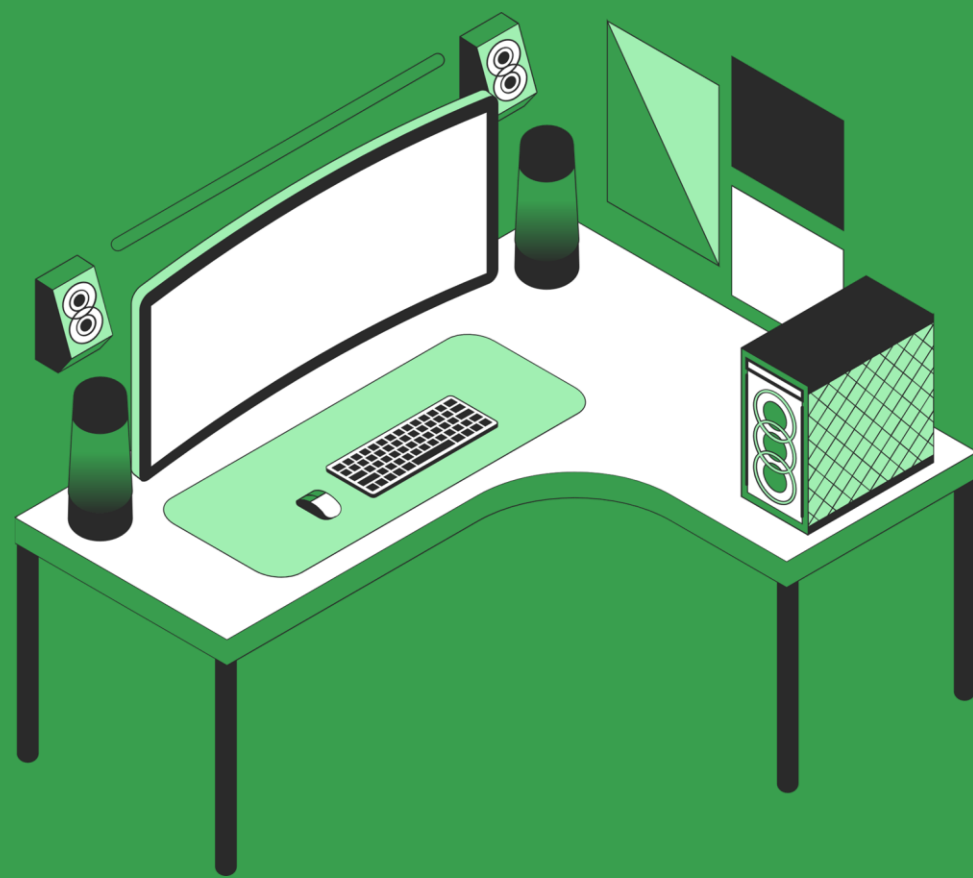
少ない工数で開発可能とはいっても、使うためのハードル、業務との相性などはツールによって異なることを再確認しました。

	楽々Framework3	CELF	PowerApps等
始めやすさ	△	○	○
操作性 学習コスト	△	○	△
柔軟性	○	△	○
業務適合性	ビジネスアプリ向け	個人・少人数向け	個人レベルも複雑なものも

未経験者がこれらを考慮してツールを選定するのは困難・・・

状況に応じて適切なツールを選ぶためのガイドが必要





### III. ノーコード/ローコード 開発スタートガイドと有用性

# スタートガイド策定



# まずはスタートガイドをお手元に

スタートガイドー表紙



スタートガイドはこちら。[ここ](#)からダウンロードも可能。





# 想定するスタートガイドの利用者



スタートガイドで、ツールの選定から開発、運用まで様々なポイントを解説



# スタートガイドの活用目的と活用方法

## 活用目的

システムに精通していない担当者が、  
ノーコード/ローコードツールを導入し業務改善ができる



活用方法：各自の悩みに沿って、本書の各省を参照する

- 利用を始めたいが何から手をつけたらよいか？ → 『[2.1 知っておきたいツールの特徴](#)』へ
- どんなツールを選定したらよいか？ → 『[2.2 ツール選定の流れ](#)』へ
- 開発に着手する前に必要なことはあるのか？ → 『[3.1 事前準備](#)』へ
- 作りっぱなしを防止したい！ → 『[4.1 管理方法](#)』へ

スタートガイドー「ガイドラインの活用目的と活用方法」

# スタートガイドの記載内容

## スタートガイドー「目次」

1. [ガイドラインの概要](#)
  - 1.1. [ノーコード/ローコードの定義](#)
  - 1.2. [歴史と背景](#)
  - 1.3. [ノーコード/ローコードの種類と導入率](#)
  - 1.4. [主な開発手法の特徴](#)
  - 1.5. [ガイドラインの活用目的と活用方法](#)
  - 1.6. [必要なスキル](#)
2. [選定](#)
  - 2.1. [知っておきたいツールの特徴](#)
  - 2.2. [ツール選定の流れ](#)
  - 2.3. [ツールの導入体制](#)
3. [開発](#)
  - 3.1. [事前準備](#)
  - 3.2. [開発](#)
  - 3.3. [テスト](#)
  - 3.4. [リリース](#)
4. [運用/展開](#)
  - 4.1. [管理方法](#)
  - 4.2. [システム改善のポイント](#)

項目のリンクをクリックすれば該当のページへ移動可能



## スタートガイドの期待される効果

初心者がノーコード/ローコードツールを **選定** できる

初心者がノーコード/ローコードツールで **開発** できる

ノーコード/ローコード開発の **野放しを防ぐ** ことができる



## スタートガイド紹介 登場人物

ここからはこの2人の掛け合いとともにスタートガイドの内容と工夫したポイントをご紹介します。



名前：ユータ (27)

説明：悩める社員  
経験浅め



名前：ユージ (35)

説明：スタートガイド所有  
ユータの先輩

経験の浅いユータ君でも、ノーコードツールの開発や運用をできるようにする！



# 想定シーン①：困っている後輩

ガイドの基本事項

読みやすさの工夫

勘所を知る工夫

開発で終わらない

ノーコード／ローコードツールを活用してみようと  
後輩が困っているが、教えている時間がない・・・



ユータ

DXのプロジェクトで「ノーコードツールを活用しろ！」  
といわれたけど、どうしたらいいんだろうか？



ユージ

ユータ君！どこから手をつけていいかわからないんだね。  
これをまず読んでみたらどうかな。

# はじめに対象者や活用シーンを記載

ガイドの基本事項

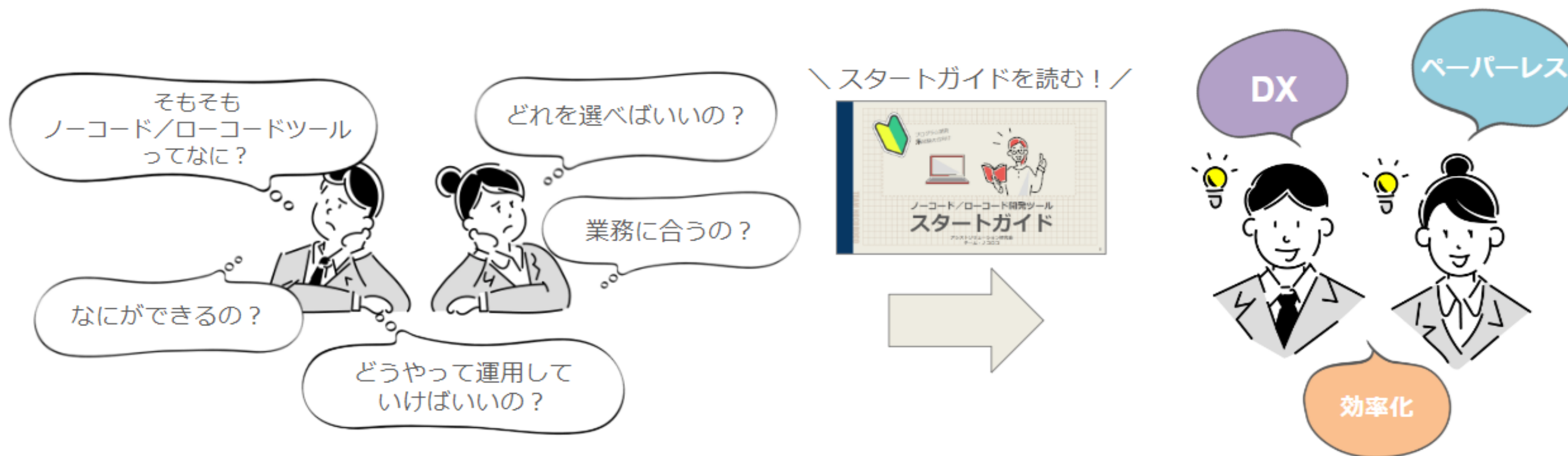
読みやすさの工夫

勘所を知る工夫

開発で終わらない

## はじめに

プログラミングの必要のないノーコードツール、プログラミングの必要が少し出てくるローコードツールはユーザー部門が自分たちでシステム開発を行うことができる夢のようなツールですが、事前に知っておくべきことやおさえておくべきポイントがいくつもあります。



このスタートガイドでは、ツールの選定から開発、運用まで様々なポイントを解説しています。  
また、専門的なIT知識を持たない担当者（組織）が、小規模の業務改善用のシステムを開発できるようになるよう

まさに僕のような人のための「スタートガイド」だね。



ユータ

目的

選定

開発

運用

# ガイドの各ページの見方

ガイドの基本事項

読みやすさの工夫

勘所を知る工夫

開発で終わらない



## ガイドラインの見方

**3.1 事前準備**

開発するためには事前に様々なことを確認しておく必要があります。ここでは主な項目について記載します。

- 1 開発する前に押さえるべきポイント**  
開発のルールについて確認する必要があります。  
開発範囲や業務運用の変化など、開発の詳細ではなく業務全体の観点で問題ないかを確認してください。
- 2 ノーコード/ローコード開発に関する全般を管理する部門（担当者）が確認する事項**  
開発を管理する部門（担当者）は、開発工程をスムーズにする役割があります。  
そのため、重複する機能を持つシステムの有無、担当者ごとの役割などを確認する必要があります。
- 3 開発手法の選択**  
開発手法についてあらかじめ決めておく必要があります。「作りながら仕様を決めていく方法」と「設計書などをすべて作成してから開発する方法」があるため、進め方を関係者で共有し開発に着手してください。

各章の詳細な説明ページの前に、内容を簡潔にまとめたページを用意しました。  
気になる内容があれば、そこから読んでいただいてもOKです。

- ノーコード** ノーコードにも当てはまる、全ての方に読んでいただきたい内容になっています。
- ローコード**
- ノーコード** ローコード向けの少し難しい内容です。システム開発について詳しく知りたい方は読んでみてください。
- ローコード**

**コラム 業務整理のためのフレームワーク**

業務整理にお役立ちなフレームワークをご紹介します。  
迷ったり、行き詰った時には参考にしてみてください。

**アンカー**  
業務内容についてアンカーを行います。  
ポイント：業務は順序とし、役割別の業務を成り立たせる必要があります。

**インタビュー**  
業務内容のヒアリングを行います。  
ポイント：管理職だけでなく担当者も取り入れて実施することです。

**プレインストーミング**  
課題や作業を整理します。  
ポイント：考えすぎず、思いつらぬ内容を整理していきましょう。

**グルーピング (MEIC)**  
課題の外観整理を見出し、グルーピングします。  
ポイント：似た業務がない (MEIC) 状態にしましょう。

**ロジックザリー**  
課題を上記の観点から下の観点に分解します。  
本質的な課題が「どこに」「なぜ」存在するか、分かるようにします。  
ポイント：課題別の関係性・因果関係に注目しましょう。

**ロードマップ**  
関係者ごとに必要な事項や実施の順序を整理して書きます。  
ポイント：具体的なアクションにつなげるため、目録も記入しましょう。

**アクションプラン**  
行動の計画を、アクションプラン書に記入します。  
ポイント：「誰が」「いつ」「どこで」「どのように」実施するのかを記載しましょう。

より詳細な内容として、コラムを用意しました。

スタートガイドの見方のページもあるよ。





## 想定シーン②：ガイドを読むことも難しい？

ガイドの基本事項

読みやすさの工夫

勘所を知る工夫

開発で終わらない

システム開発未経験者にとって、  
このガイドを読むことも難しいのではないかな？



ユータ

読むのはよいですが、未経験者の僕だとボリュームも多いし難しい  
のでは？どこが重要なポイントなのか、わからないのではないですか？



ユージ

そうだね。でも大丈夫！読みやすくするための工夫もしてあるんだ。  
これを見てごらん。

## 2.2 ツール選定の流れ

様々なツールが存在する中で、業務課題の解決が可能なツールを選択するのは非常に大変です。  
ここでは、ツールを選定するまでの流れについて説明します

1

### 業務課題の整理

ツールのことは一旦脇に置いて、まずは現状の業務課題を把握することから始めることを推奨します。

2

### ツール適用範囲の検討

解決したい課題が何かを把握したところで、課題を解決するために業務のどの部分にツールを適用すればいいのかを検討します。

3

### ツールの選定

ツールを利用したい業務を絞り込んだところで、ツールの選定に入ります。

#### 選定フロー

導入事例へのリンクを含む選定フローを用意しました。

ツール適用範囲の検討、ツールの選定の参考にしてみてください。

選定や開発などの全体順序があって、詳細の内容を説明しているよ。





## ☕ コラム 業務整理のためのフレームワーク



業務整理にお役立ちなフレームワークをご紹介します。  
迷ったり、行き詰ったりした時には参考にしてみてください。

### アンケート

業務内容についてアンケートを行います。

ポイント：質問は簡易にし、回答者の負担を減らすと回答を得やすいです。

### インタビュー

業務内容のヒアリングを行います。

ポイント：管理職だけではなく担当者を含めて実施すると良いでしょう。

### ブレインストーミング

課題を付箋に書き出します。

ポイント：考えすぎず、思いつくものを早く書き出していきます。

### グルーピング (MECE)

課題の共通事項を見出して、グルーピングします。

ポイント：抜け漏れがない (MECE) 状態にしましょう。

### ロジックツリー

課題を上位概念と下位概念に分解します。

本質的な問題が「どこに」「なぜ」存在するか、分かるようにします。

ポイント：課題間の関係性・因果関係に注目しましょう。

### ロードマップ

目標達成に必要な事項や困難な事柄を時系列で書き出します。

ポイント：具体的なアクションにつなげるため、日付を記入しましょう。

### アクションプラン

行動の詳細を、アクションプラン表に記入します。

ポイント：「誰が」「いつ」「なにを」「どのくらい」実施するのか？を記載しましょう。

目的

選定

開発

運用

知っておくと便利なことはコラムにしてあるよ



## 想定シーン③：ガイドのどこまでやるべき？

ガイドの基本事項

読みやすさの工夫

勘所を知る工夫

開発で終わらない

スタートガイドに記載してあることのどこまでをやったらよいかかわからず、諦めてしまっているのではないかな？



ユータ

業務のどの部分をツールで自動化するかチームで検討しててもきれいに整理できなくて。どうしたらいいんですか？



ユージ

ツールの適用範囲を決めるのは重要だよ。でも、やり過ぎても良くないこともあるから曖昧さも必要だよ。

## 2.2.2 ツール適用範囲の検討

業務課題を解決するために、業務のどの部分にツールを適用するかを検討します。  
ただし、ツールにより対応可能なことが異なるため、ガチガチに適用範囲を決めることは推奨しません。  
判断基準例を以下に記載します。



### 向いている業務

- 定型業務である
- 単純作業だが、時間が掛かっている
- 紙やExcelで運用している
- 難易度が低いが、業務改善効果が得やすい
- 複数システムへの重複入力が発生している
- 本社と現場など遠隔地での情報共有に課題がある
- システム間のデータ連携

### 向いていない業務

- 実現したい内容が複雑
- 求められる機能が多い
- 堅牢なセキュリティが求められる



ツールにより適用範囲が変わる場合があるよ

概要

選定

開発

運用



# 押さえておきたい観点は網羅

ガイドの基本事項

読みやすさの工夫

勘所を知る工夫

開発で終わらない

## 3.3.2 テストの実行

ノーコード

ローコード

正常系・異常系

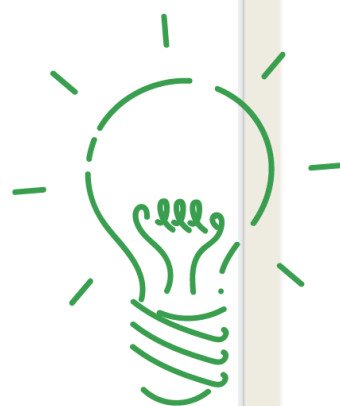
	観点	サンプル条件	サンプル結果
正常系	想定された値が入ってきた時、想定した結果になること	項目「メールアドレス」に「hoge@example.com」を入力する	正常に登録されること
	想定された操作がされた時、想定した処理が実行されること	存在するアカウント情報「ID」と「パスワード」を入力して、ログインボタンを押下する	正常にログインできること
異常系	想定されていない値が入ってきた時、想定した結果になること	項目「メールアドレス」に「山田太郎」を入力する	登録されず、エラーメッセージが表示されること
	想定されていない操作がされた時、想定した処理が実行されること	存在しないアカウント情報「ID」と「パスワード」を入力して、ログインボタンを押下する	ログインはできず、エラーメッセージが表示されること

目的

選定

開発

運用



ユータ

正常なデータ以外もテストしないといけないってことか。  
テストの項目で漏れてしまうところだった。

## 想定シーン④：開発したら終わり？

ガイドの基本事項

読みやすさの工夫

勘所を知る工夫

開発で終わらない

システム開発 = ゴールとなり、その後の対応が疎かに . . .



ユータ

ようやく開発してリリースしました！  
これでプロジェクトは終了ですね。ありがとうございました。



ユージ

いやいや、ユータ君。これからも運用保守はおこなっていかないと  
いけないよ。スタートガイドにも書いてあるでしょ？

## 4.1 管理方法

この章では、ノーコード/ローコード開発を安全に進めるために、また、ガバナンスを機能させながら内製の効果をより得ることができるように、管理が必要となる事項について記載していきます。

- 1 開発時の推奨事項
- 2 保守体制
- 3 障害発生時の対応
- 4 ライセンス管理
- 5 セキュリティ
- 6 バージョンアップ
- 7 バックアップ



ユータ

障害発生時の対応やライセンス管理、セキュリティ！？  
確かにトラブルが起こってからでは遅いですもんね。

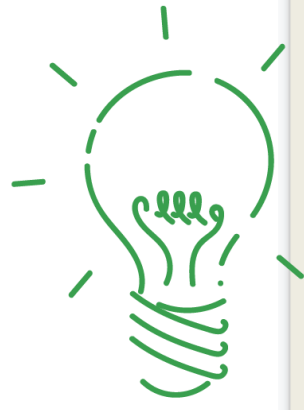
目的

選定

開発

運用





## 4.2.3 利用促進

ノーコード

ローコード

ツールの利活用を促進するため、社内でコミュニティや意見交換の場を設けるといいでしょう。業務改善を実現した事例の共有／発信を積極的に行いましょう。



### 他部門への展開時に気を付けること

#### 契約の見直し

利用者数：追加利用者分のライセンスが足りない場合  
追加で契約する必要があります。

性能契約：利用者増加に伴い負荷が増加する場合  
対応できる性能に変更する必要があります。

#### 問合せ先

サポートへの問合せ回数に制限がある場合があります。  
一部の利用者が使い切ることがないよう監視／制限をする必要があります。  
サポートへの問合せ窓口が一本となるよう体制を整えましょう。

#### 利用を許可する範囲の検討

すべてのメンバーがすべての機能を利用できるようにすることは、セキュリティリスクにつながります。  
システムの利用範囲に対して、制限をかける必要があります。

#### アクセス権限

部外秘の情報など、他部門に見せられないデータがある場合は、アクセス権を設定しましょう。  
組織／ユーザー単位で、閲覧／編集可能な範囲を設定する必要があります。

自分たちだけでなく他のメンバーにも教えていこうね

目的

選定

開発

運用



# 作成したスタート ガイドの有用性検証



# 有用性検証の概要

## 目的

1. 作成したスタートガイドの有用性を確認
2. フィードバックからスタートガイドをブラッシュアップ

## 概要

- 実施概要
  - 対象者にスタートガイドを読んでもらい、アンケート形式で評価
- 対象
  - システム開発未経験者中心
  - メンバー各自の社内や、取引先など
- 目標有効件数
  - 30件

## 主な確認事項

- 回答者の属性
  - 業種
  - 経験
  - ノーコード/ローコードの認知度
- 内容
  - スタートガイドの内容の理解度
  - 活用への期待度の評価

## アンケート回答者

45名の方へ依頼し **35** 名の方から回答

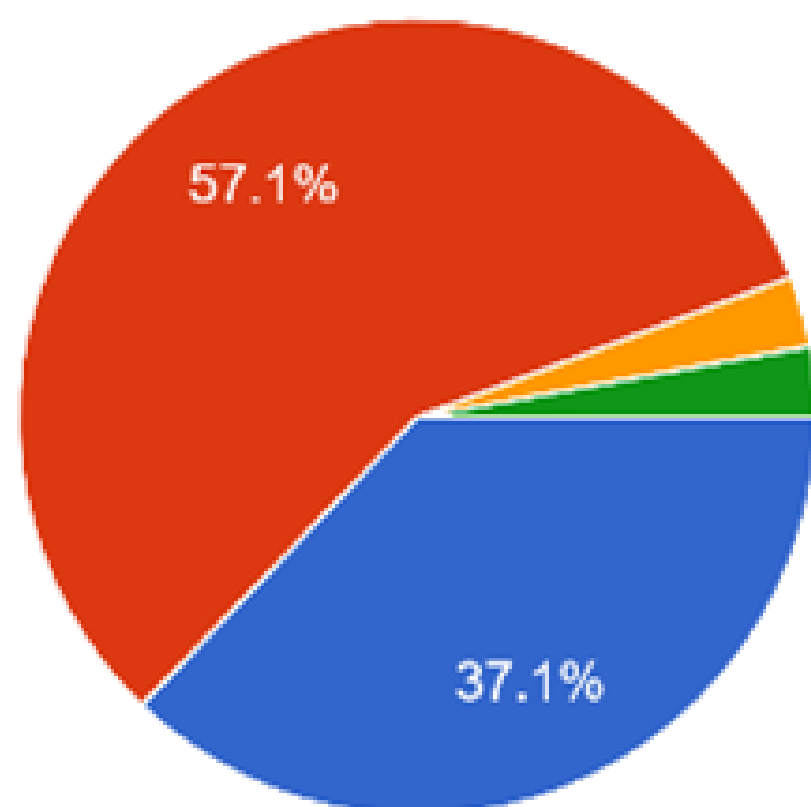
(アンケート依頼先の内訳)

社名	対象者	依頼人数
リコージャパン株式会社	営業部門、IT部門 など	5名
トランコムITS株式会社	総務部門、IT部門 など	5名
鈴与システムテクノロジー株式会社	IT部門だが開発経験無し など	6名
愛知県経済農業協同組合連合会	営業部門 など	10名
株式会社メイテツコム	IT部門、常駐先（某放送局）など	8名
株式会社アシスト	広報部門、新事業部門、営業支援部門、製品技術など	5名
アシスト取引先	某自動車メーカー、製造業、物流業	6名

# アンケート結果（属性）①

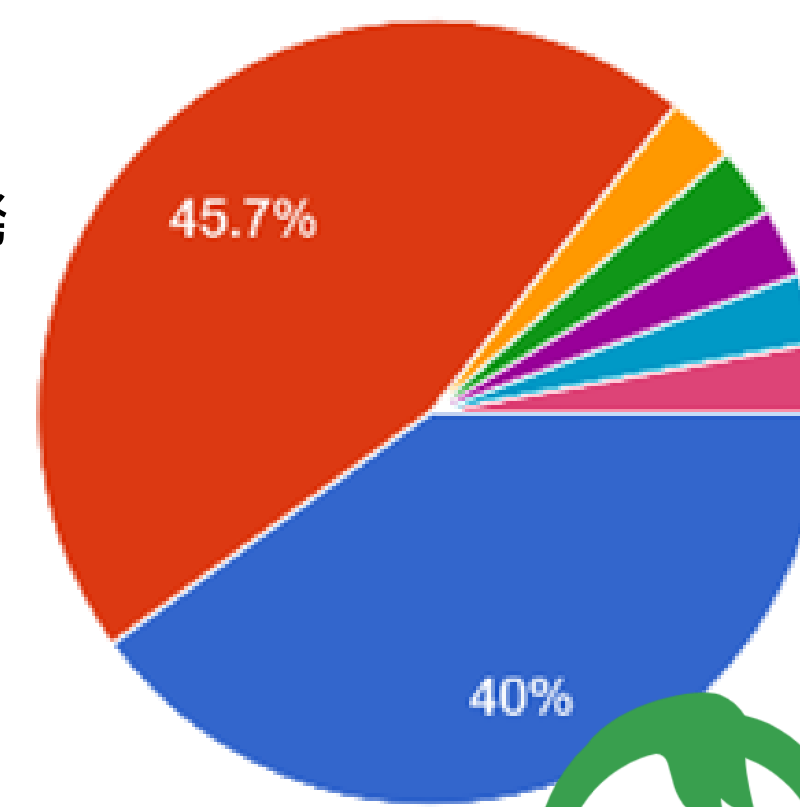
(1) 業種はどちらですか（35件の回答）

- 製造業や流通業、金融業などの非IT系
- Slerやシステム開発会社、システム子会社などIT系
- 放送業
- メディア



(2) 所属部門はどちらですか（35件の回答）

- 営業部門や人事部門など非IT系部門
- 情報システム部などIT部門
- デジタルメディア運用とソリューション開発
- イベント事業
- 新商材・サービス開発
- 生産技術部門
- 製品開発部門



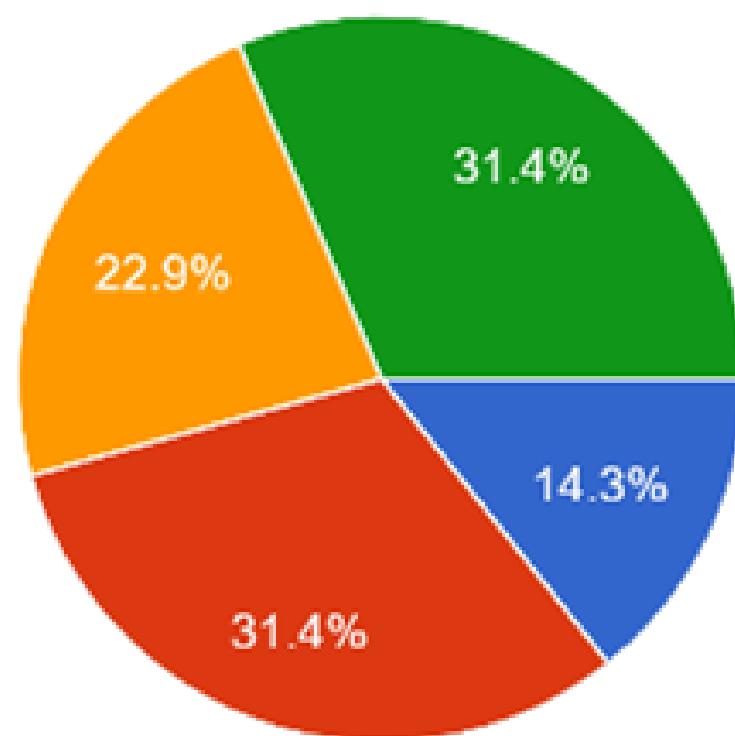
回答者のうち半数近く、システム開発と関係性が低い



## アンケート結果（属性）①

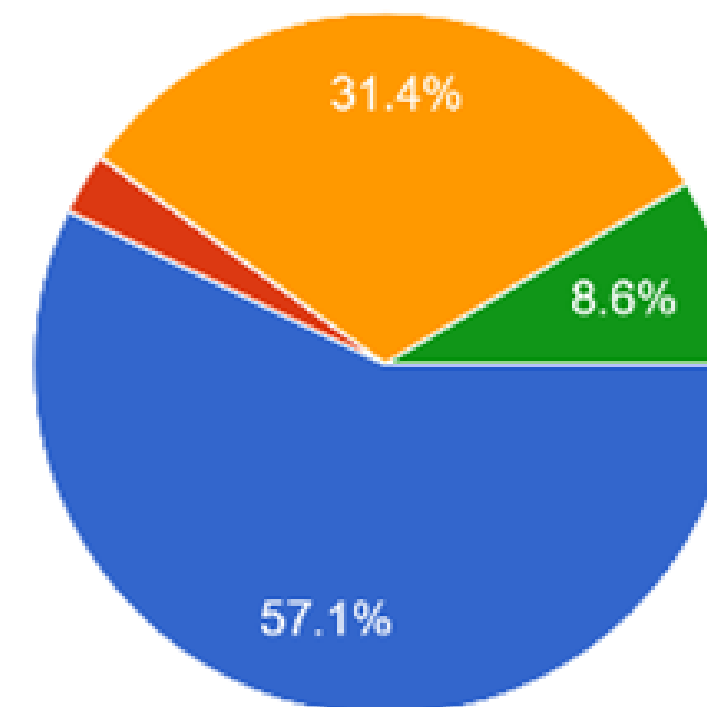
(3) ノーコード/ローコードツールを知っていますか（35件の回答）

- まったく知らない
- どういったものかは知っている
- それらツールを見たことがある。触ったことがある
- それらツールを日常的に見ている。触っている



(4) ノーコード/ローコードツールでの開発をしたことがありますか（35件の回答）

- まったく開発をしたことがない
- 開発をしようとしたが挫折した
- 数回、開発をしたことがある
- 日常的に開発をしている

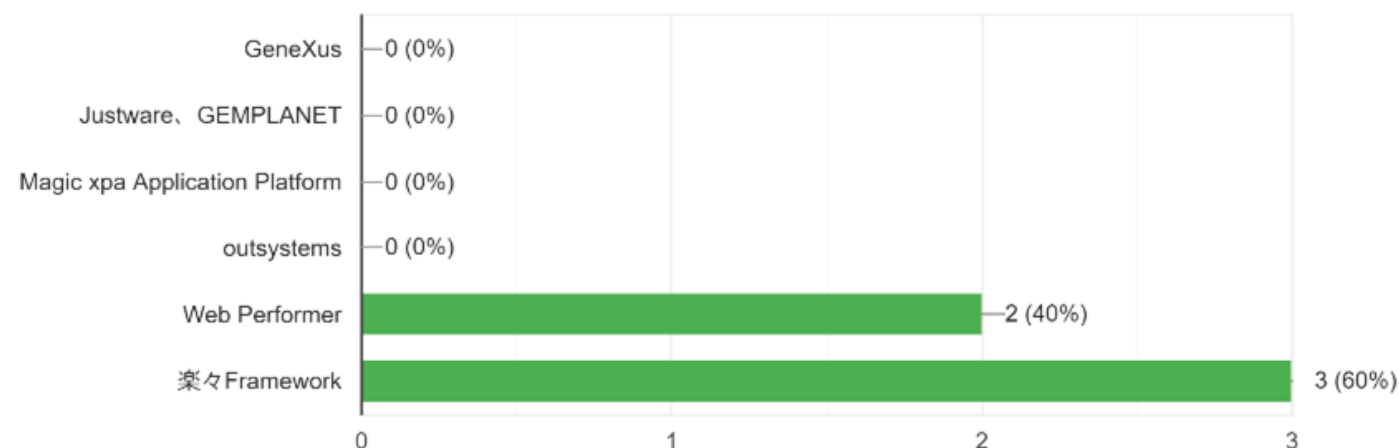


ノーコード/ローコードツールの認知度は8割以上、しかし開発未経験が約6割



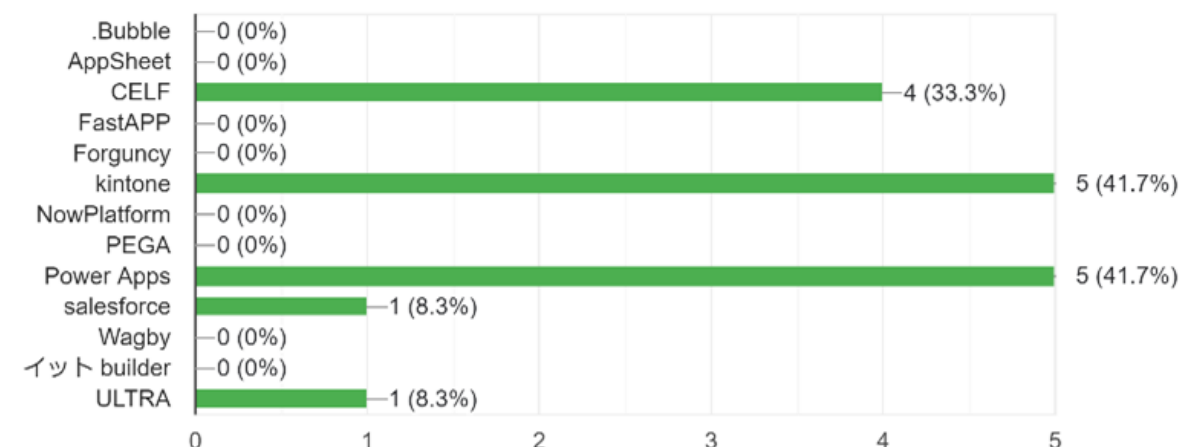
## アンケート結果（属性）②

(1) 使用経験のあるビジネスアプリ開発ツール  
（市場シェア上位）  
（複数選択可）



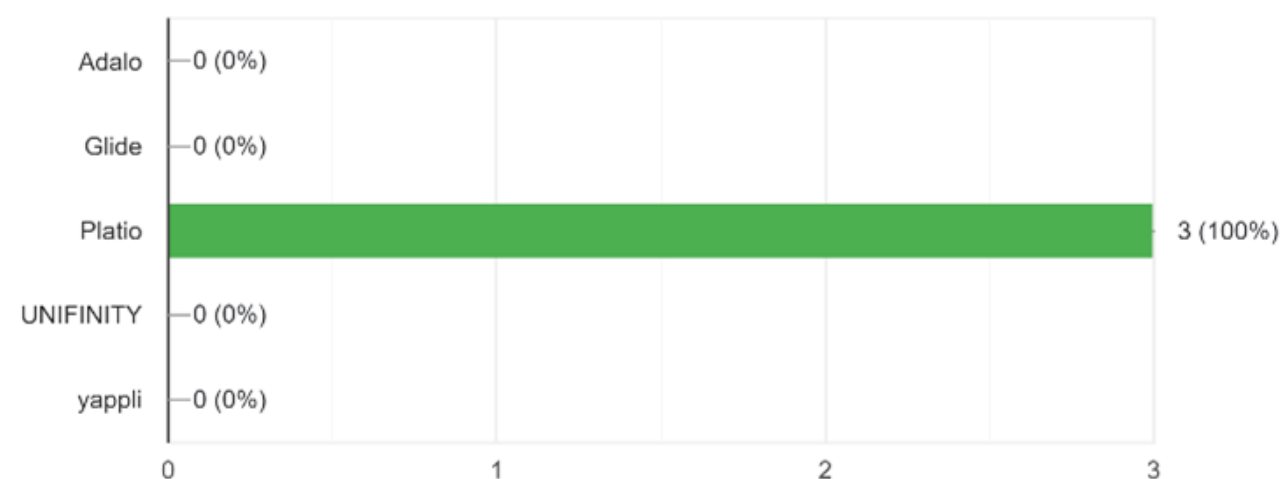
5人

(2) 使用経験のあるビジネスアプリ開発ツール  
（その他）  
（複数選択可）



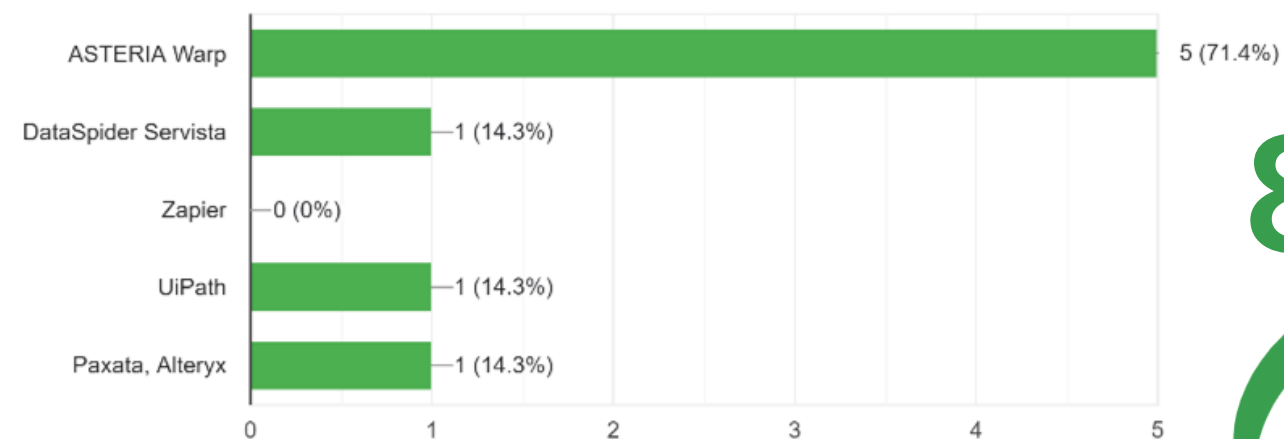
16人

(3) 使用経験のあるモバイルアプリ開発ツール  
（複数選択可）



3人

(4) 使用経験のある業務自動化ツール  
（複数選択可）



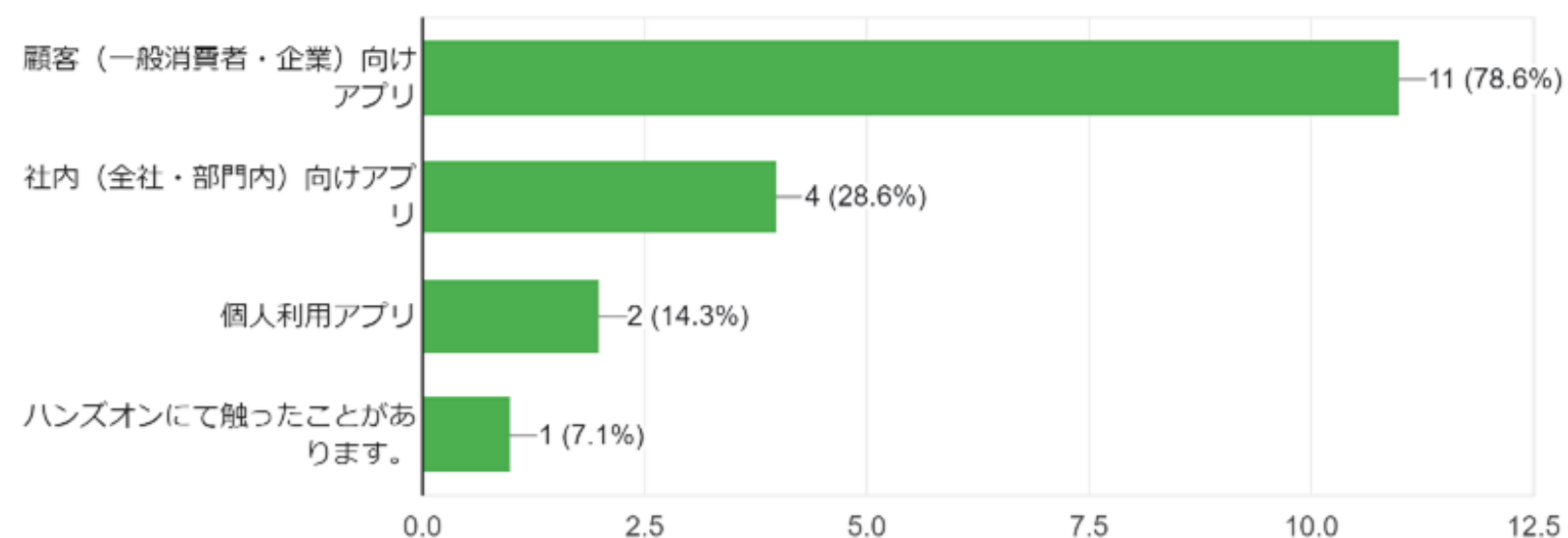
8人

ノーコード/ローコードツールを使用したことがある回答者は半数にも満たない

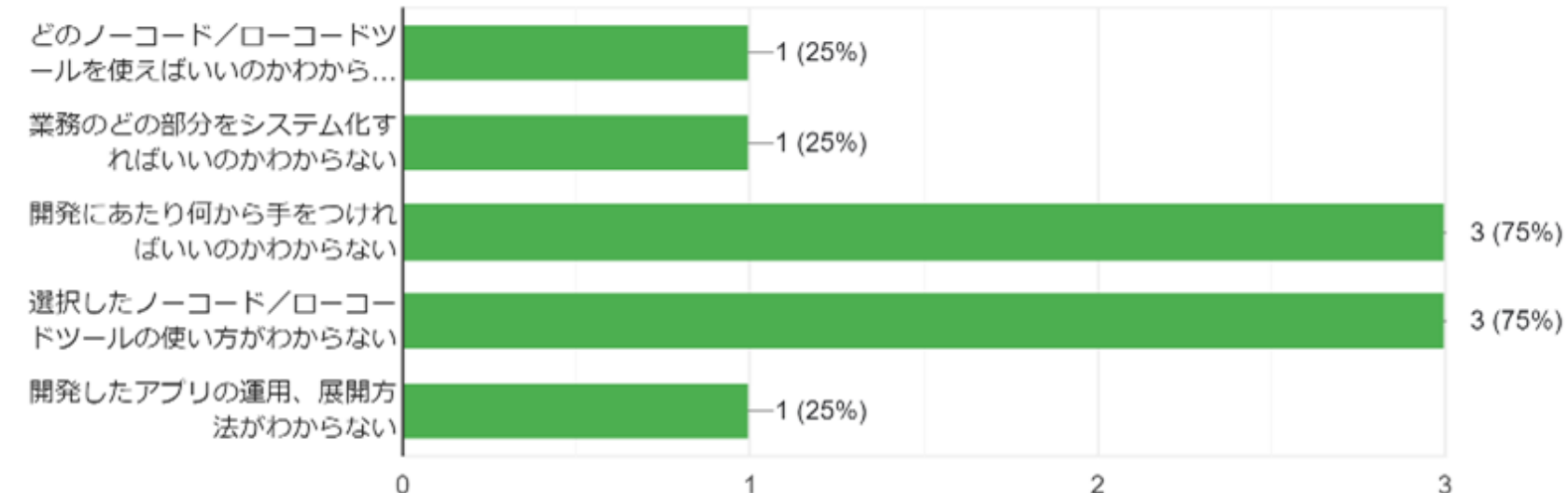


## アンケート結果（属性）③

(1) どんなアプリを開発したか（経験者のみ）  
（複数選択可）



(2) 開発途中で挫折した理由（経験者のみ）  
（複数選択可）



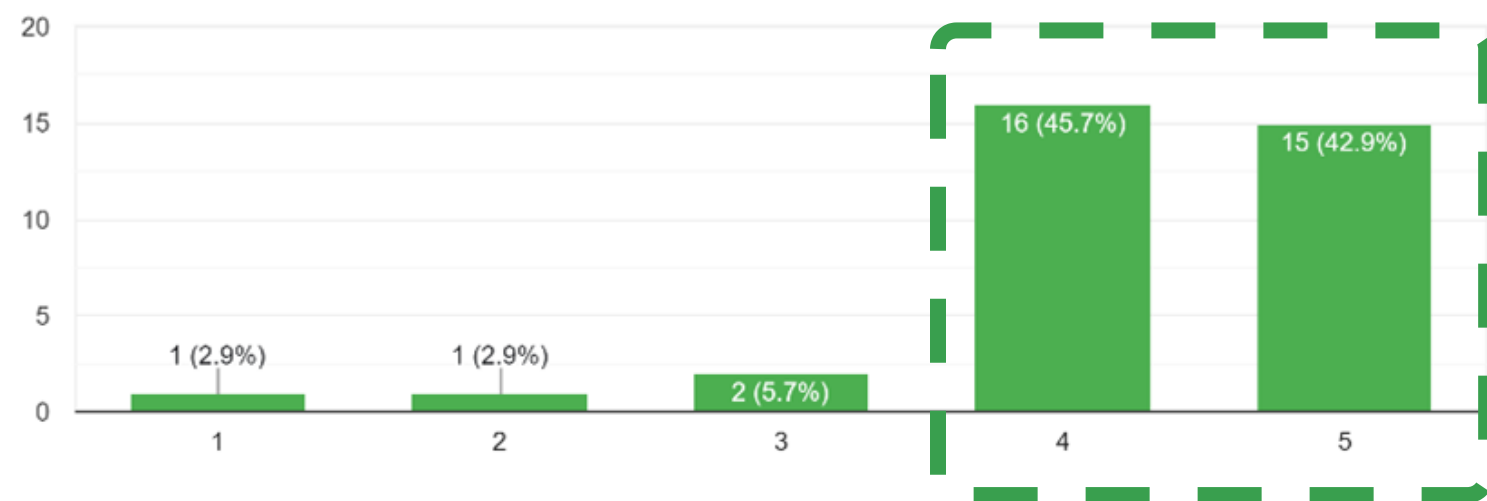
選定や開発、運用それぞれで迷い挫折している



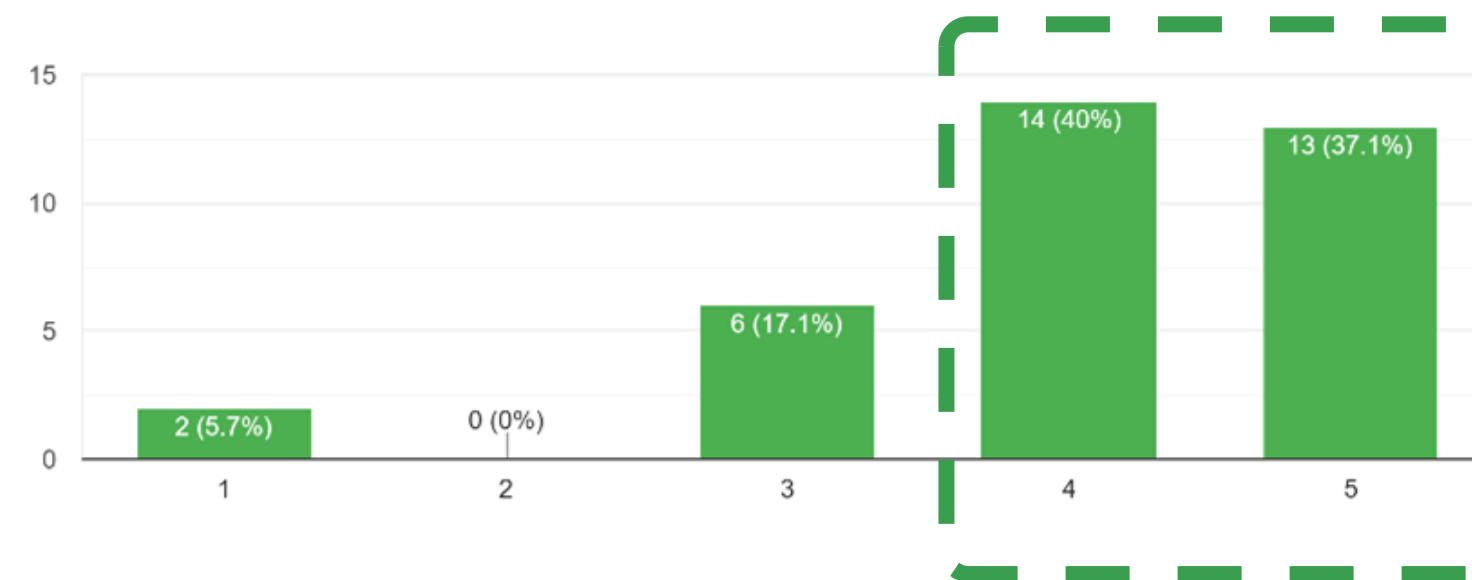


# アンケート結果（内容理解）

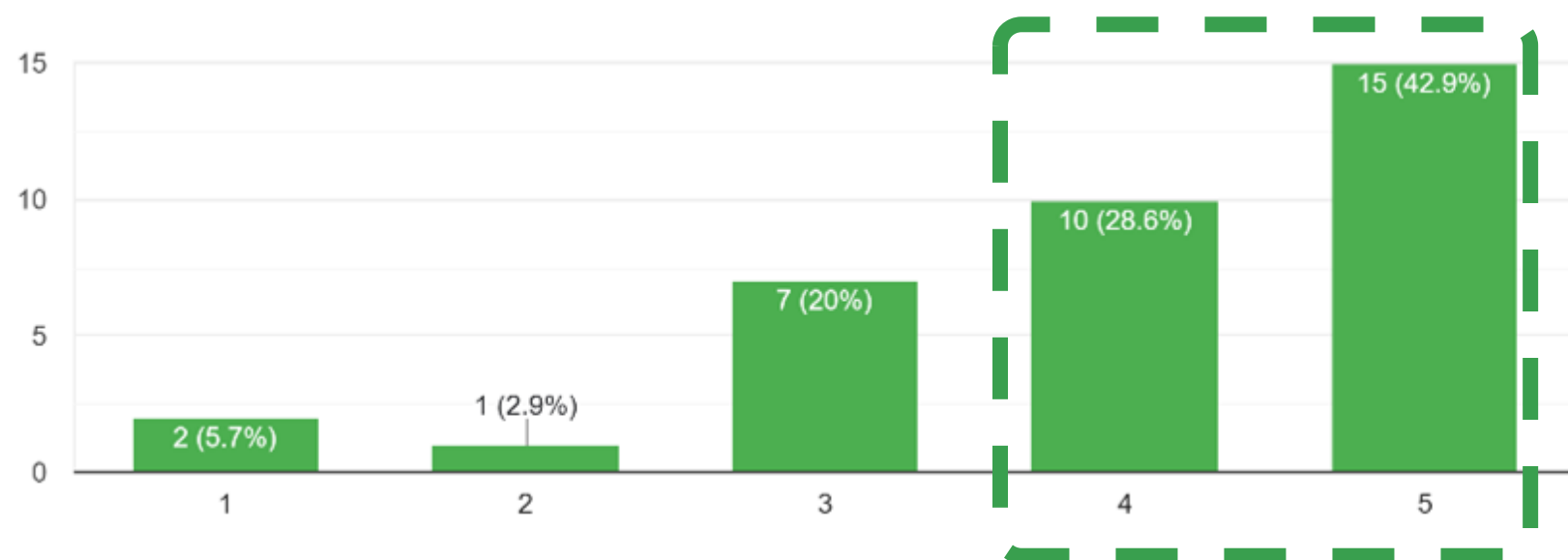
(1) 「ガイドラインの目的」の内容は理解できましたか？



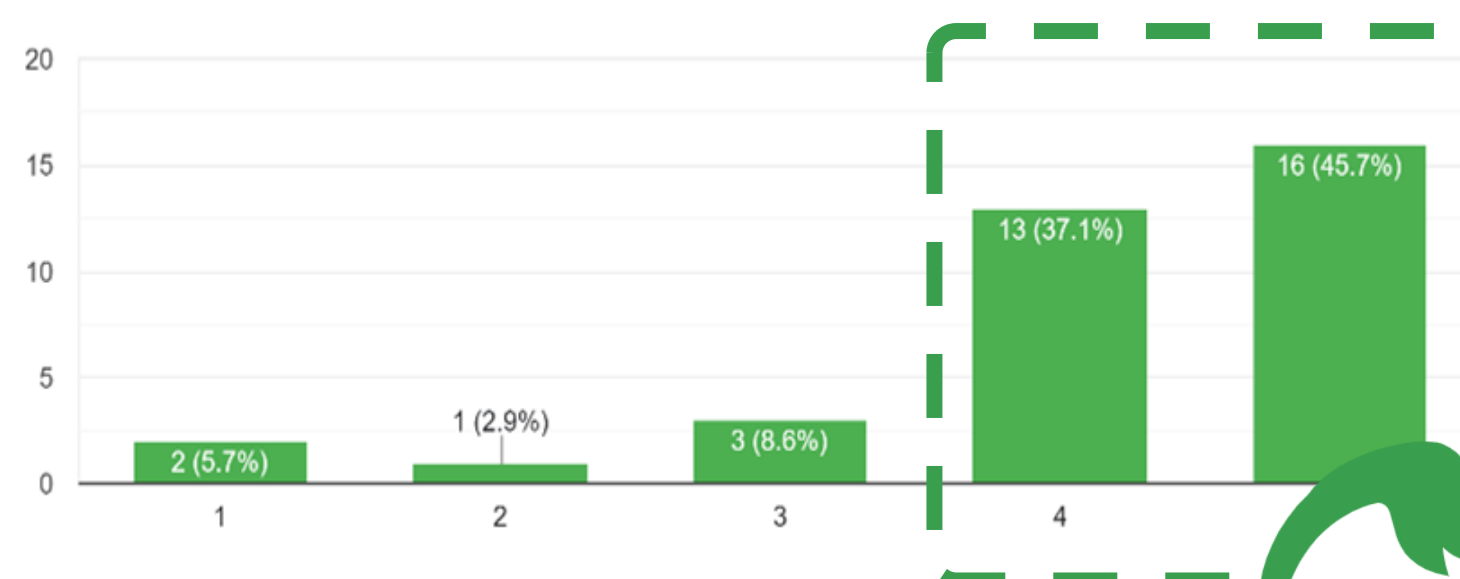
(2) 「選定」の内容は理解できましたか？



(3) 「開発」の内容は理解できましたか？



(4) 「運用・展開」の内容は理解できましたか？

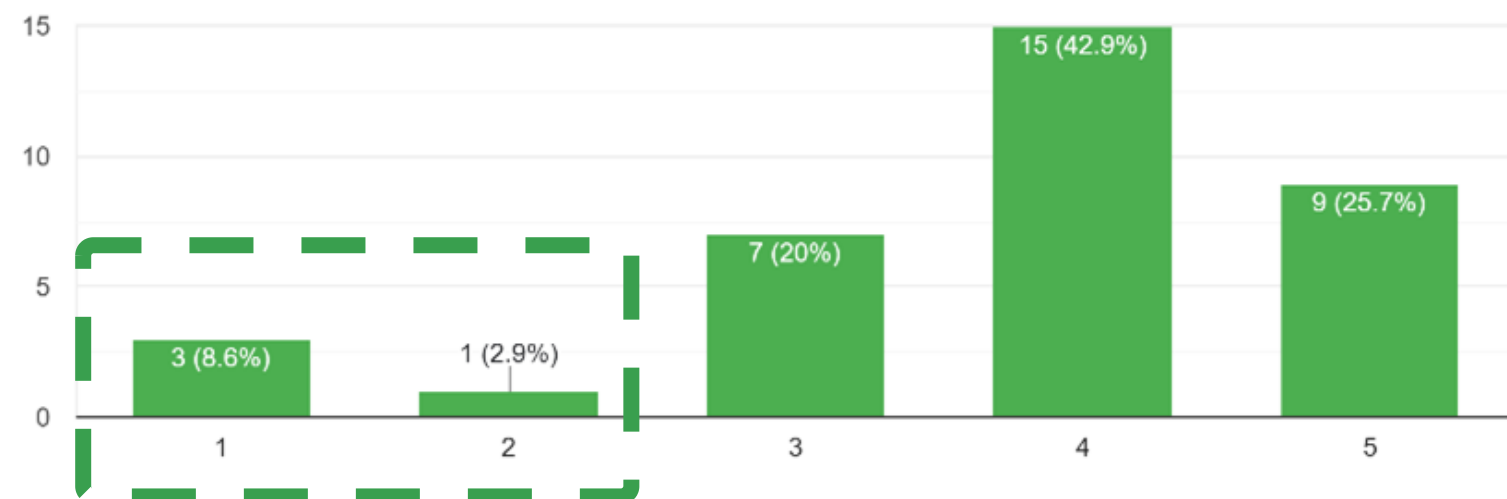


内容の理解度は非常に高い

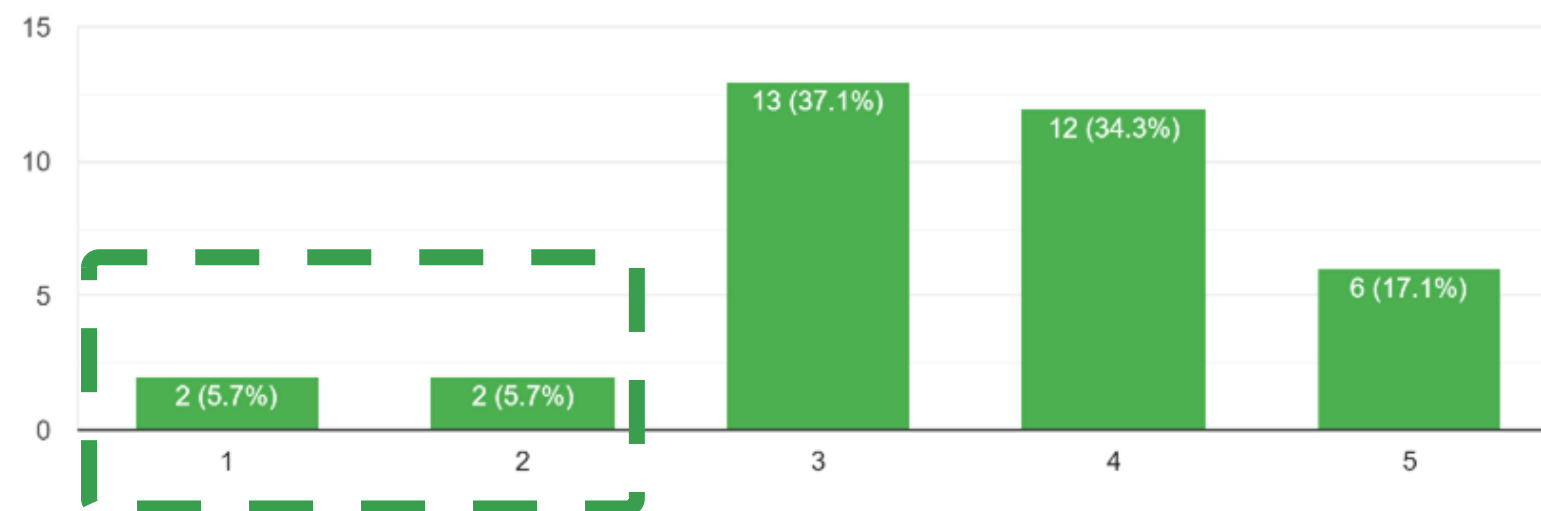


# アンケート結果（活用できそうか）

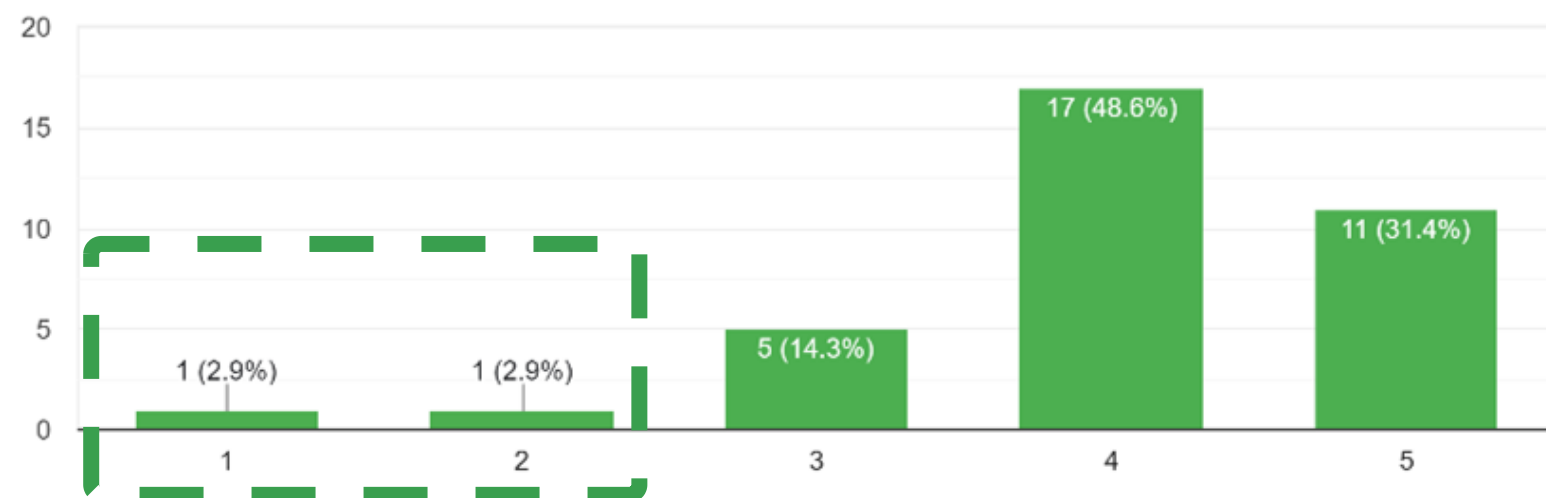
(1) 「選定」の内容は活用できそうですか？



(2) 「開発」の内容は活用できそうですか？



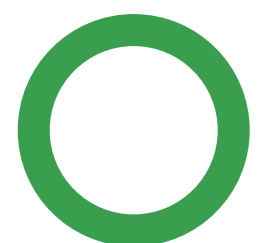
(3) 「運用・展開」の内容は活用できそうですか？



活用見込みは、低評価が1割～2割ほど



## 回答者からのコメント



### 高評価、ポジティブなコメント

- サンプルの記載もありわかりやすかった。
- 業務整理は重要でその部分のプロセスが記載されていて参考になった。
- キーとなる工程が整理されており、システム開発で検討するポイントを知る機会となった。



### ご意見、ご指摘いただいたポイント

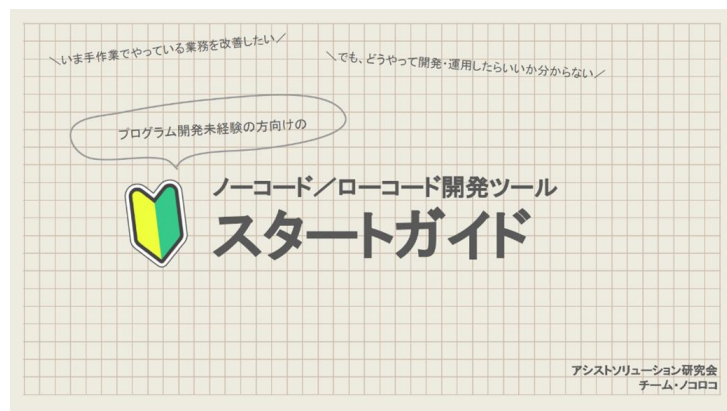
- ローコードまで考えると必要だと思うが、ノーコードをやりたい人には内容が重たい。
- 準備ややることが多く、「やることが変わっただけで忙しさは変わらないのではないか？」という気持ちになりそう。
- 理解できたが、どんなことができるのかといったところまで想像がつかない。

内容の網羅性は評価が高い。一方、内容が重たくなりハードルが高い印象



# アンケート結果からの改善

(初版)



(1.1版、1.2版)



文字を減らして  
読みやすく、  
イラストを入れて  
わかりやすく  
・・・等

### 3.2.4 データベースの設定(1)

データベースとは、システムにおいてデータを貯めておくためのツールです。ツールによって、決められたデータベースしか使用できないもの、データベースの選択が必要なものがあります。この章では、「リレーショナルデータベース(RDB)(※)」と呼ばれるデータベースについて説明します。  
※Excelの表のような形をとったデータベース。行と列で構成される。  
【代表的なRDB】PostgreSQL, MySQL, OracleDBなど

データベースの設定は、後から変更するのが難しい(もしくは不可能な場合も)です。また、テーブル構成によって、処理効率に大きな影響を与える場合があります。そのため、設計段階で慎重に検討する必要があります。

**テーブル構成**

- 正規化  
一つの要素から他の要素が決定される場合、別のテーブルに分ける必要があります。これを正規化といいます。第3正規形、またはあえて正規化しない非正規形にすることが多いです。非正規形に向いているのは、Webのアクセスログなどのログ系に多いです。  
→ 詳しくは『コラム「正規化」とは』へ
- 複数テーブルでのデータの紐づけ(外部キーの活用)  
外部キーとして別テーブルの値を利用すると、データ探査の処理効率が上がったり、データが存在しない場合にエラーとして検知することができます。

40

### 3.2.4 データベースの設定(1)

データベースとは、システムにおいてデータを貯めておくためのツールです。ツールによって、決められたデータベースしか使用できないもの、データベースの選択が必要なものがあります。本スタートガイドでは「リレーショナルデータベース(RDB)」を前提に説明をすすめていきます。

リレーショナルデータベース(RDB)とは

Excelの表のような形をとったデータベースで、行と列で構成されます。1つの表をテーブルと言います。また、複数のテーブル同士の関係を定義することで、複雑なデータの組み合わせも扱うことができます。高度な検索を行うことができるのが特徴です。一方で、データを扱う場合は、件数が多いほど処理に時間がかかるため、Webのアクセスログなどの大量にあるデータの扱いには向きません。

【代表的なRDB】  
PostgreSQL, MySQL, SQL Server, OracleDBなど

アンケートでのご意見もふまえ、60項目以上の改善を実施



# 研究結果

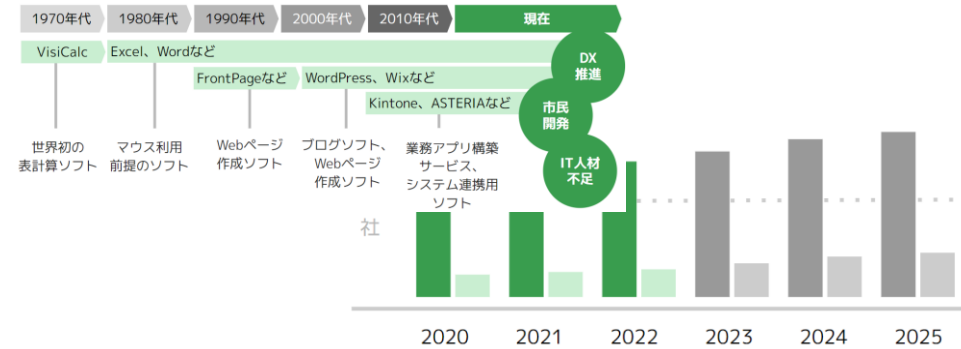


# 研究結果まとめ

私たちは当初から意思決定手法のOODAループを意識して活動を進めて参りました。

Observe  
観察

市場動向  
調査



8

つのツールを触ってみた

Orient  
情勢適応

1つの  
業務で  
ツール比較

3

チームで比較してみた

問題点：難しい・野放し

課題： 選定 開発 運用

Decide  
意思決定

研究テ  
ーマを  
決定

「ノーコード/ローコードツールのスタートガイド策定」  
ツール選定だけでなく開発・運用も含めた勘所をスタートガイドとして  
まとめてリリースし、その有用性を検証する

仮説

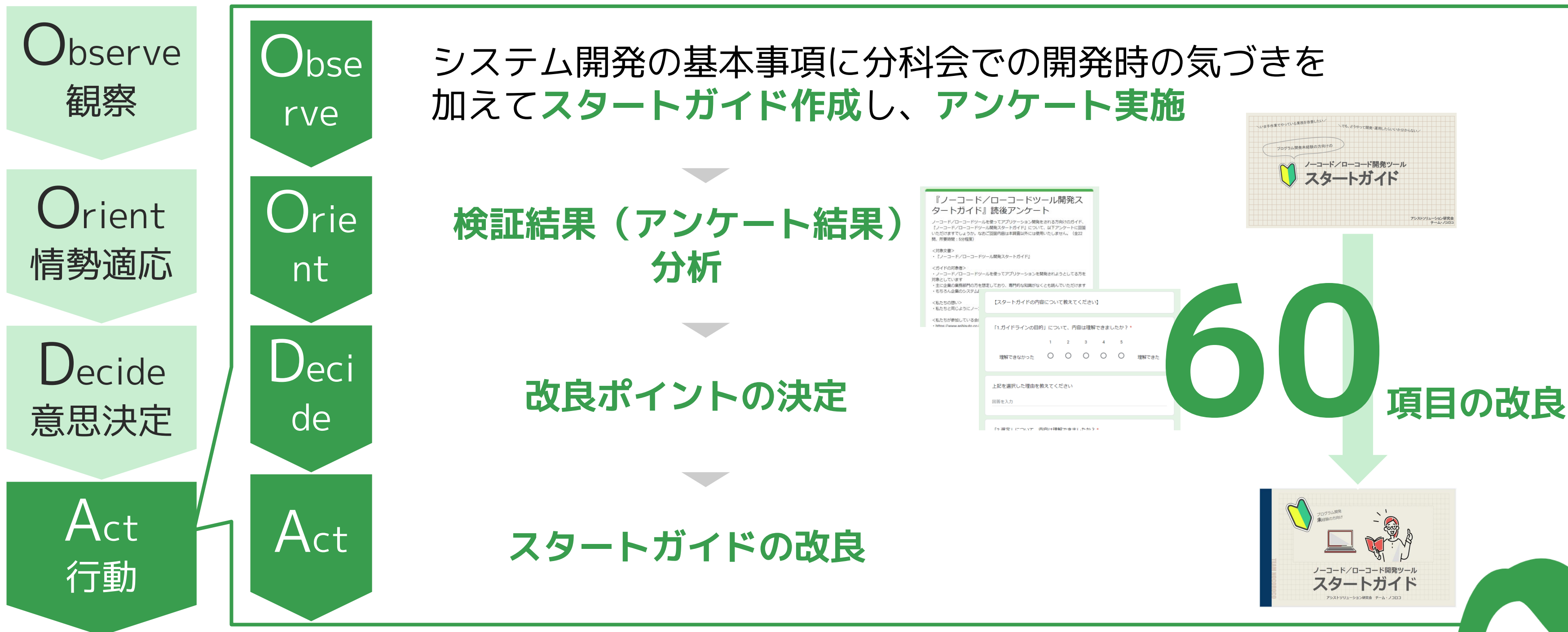
Act  
行動

「観察」「情勢適応」から問題点を抽出して課題を設定  
課題を解決する仮説を、研究テーマとして「決定」



# 研究結果まとめ

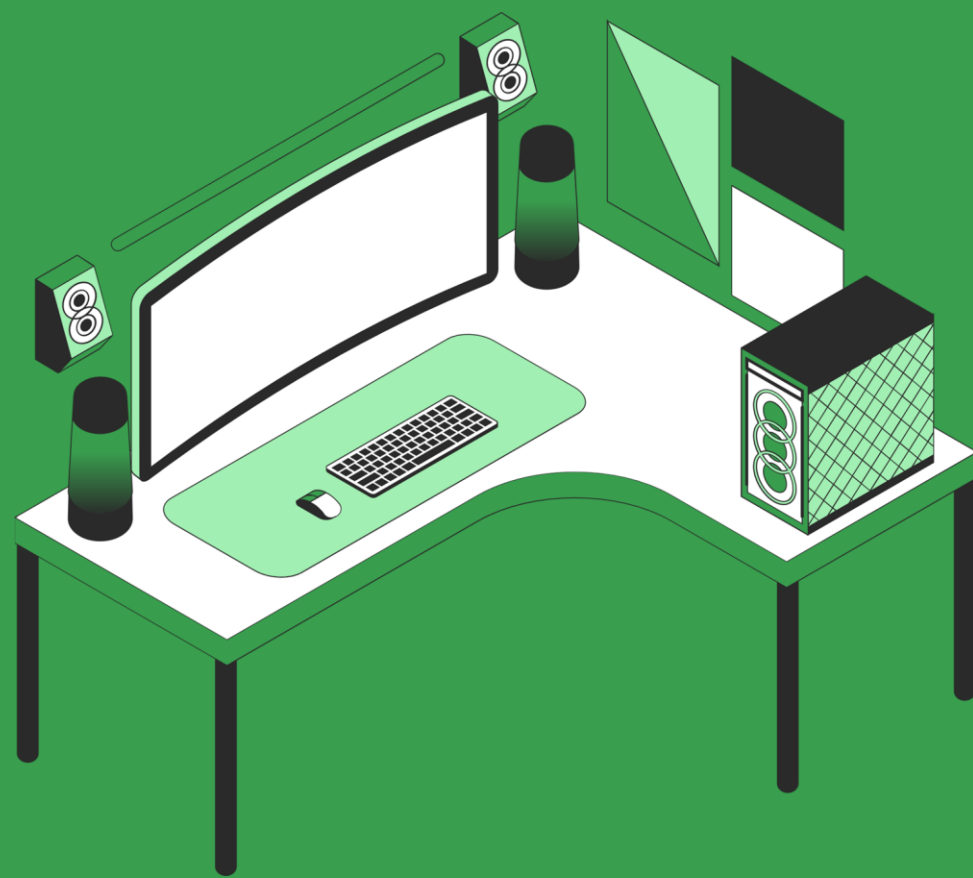
「行動」ステップの中でも再度OODAループを進めました。



スタートガイドは、未経験者の「選定」と「開発」の手助けとなり  
また「野放し」に対しても一定の道標を示すことができた



# IV. 総括





# 総括



# 総括

## メンバの多様性を活かした議論

メンバーの立場や開発経験、知識のばらつきがあったからこそ、どこまで記載すべきか、どういった用語にすべきかの議論ができ、結果とし読み手にとってわかりやすいガイドができました。

## OODAループを経てリアリティのあるガイドを作成

OODAループ（見る、わかる、決める、動く）をベースに、自分たちで実際に開発を行ったことは、スタートガイドにおいてリアリティを埋め込むことができたと考えています。

## 仮説の立証

スタートガイドが有用であると評価いただいたことで、ガイドによって未経験者であってもノーコード/ローコード開発ができるだろう、という仮説を立証できたと考えています。

## メンバ所感



青木

この**活動で得た教訓は今後PJを進めていく上で活かせるものが多く**、非常にありがたい経験をさせていただいたなと感じております。

今回作成したスタートアップガイドは、**ノーコード/ローコード開発に十分活用できる**と思います。



武田

ガイドの表現方法や内容の精査には苦勞しましたが、**最終的にはメンバーの想いが一致した**良い活動ができたと思います。

メンバーの皆さんと時にぶつかりながらも活動し、**最後まで悩みぬいただけあって満足**のいくものが出来たと思います。



森山



小久保

## メンバー所感



田中

ミッションの思いをメンバー間で共有でき、気持ちを途切れさせず活動できたのが良かったと思います。

どのようなポイントを押さえ、開発を進めていけば良いのか？「**本当に活用できる、嬉しいシステム**」の実現に**ガイドを活用**いただけると幸甚です。



宮澤



松尾

ガイドを作って終わりではなく**分科会から外へ出て声を集めその有用性を確かめる**ところまでこだわることができたこのチーム最高です。

# 展望



# そして、スタートガイドの今

スタートガイドを策定しその有用性も検証しましたが、検証に協力いただいた皆さんの声をふまえると、もっと良いものしたいとの思いが芽生え・・・

実はその後もスタートガイドの改訂を続けました。



## やまない私たちの想い

加えて、当初掲げた私たちのミッションの実現に向けて、以下も進めつつあります。

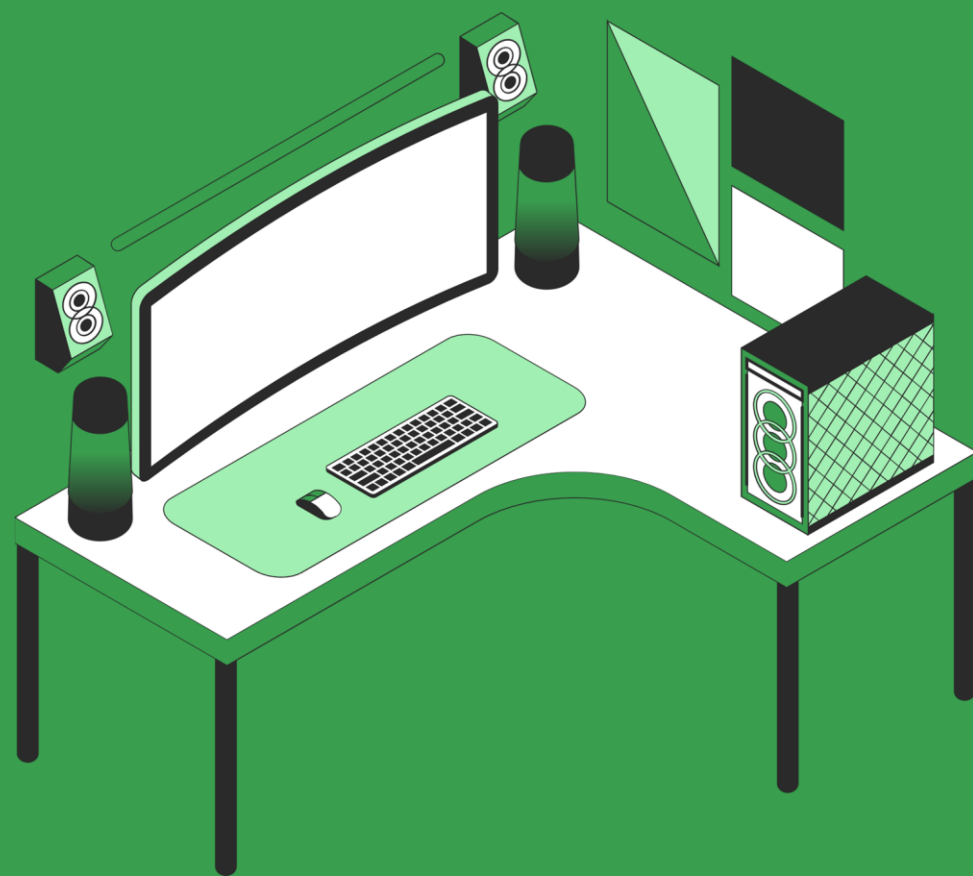
スタート  
ガイド  
ウェブ公開

所属企業で  
スタート  
ガイド  
勉強会

スタート  
ガイド  
説明動画  
社内公開

“ 本分科会で研究した内容をガイドラインとしてアウトプットし、私達と同じようにノーコード／ローコードツール活用に悩める人達に貢献したい！

ご清聴ありがとうございました



チーム・ノコロコ