

「仮説検証」と「改善」で変化の激しい時代を乗りこなせ！

# 「アジャイル開発」 導入ガイド



株式会社アジャイルウェア

スクラムマスター / アジャイルコーチ

Team.KAIZEN 組織マネジメント / 開発マネジメント

## 平川 隆仁

製造業界でのエンジニア職およびスクラムマスターの経験を経てアジャイル開発の奥深さを知る。組織体制にとらわれず、チーム本来の生産性を発揮できるワクワク組織を作るためにアジャイルウェアに入社。Lychee Redmineのプロダクトオーナーや複数チームのスクラムマスターを歴任。企業規模の拡大に伴い Team. KAIZEN を立ち上げ、現在は全社のカイゼン活動をリードしながら、より働きやすく成果を上げられる組織の実現を追求している。

その他のコミュニティ活動: スクラムフェス大阪 実行委員



# アジャイルはなぜ必要？

これまでの仕事の仕方と何が違う？

# 変化が早く正解がわかりにくい時代のアプローチ

## 過去

物が少なく、作れば売れる。売り切りサービスが中心。  
生活を便利にするものは売れ、大量生産して薄利多売に。  
作るものを決めたら、早く安く作ることがバリュー。  
量産できる設備と物流網でスピードを上げコストを落とす。



## 近年の傾向

サービスも消費者も価値観も多様化。SaaS サービス中心。  
何を作れば売れるのかがわかりにくい反面、  
小さく早く市場に出し、徐々に改善・成長させやすい。  
物理的な大規模投資も大型の物流網も不要。



変化に適応したビジネスだけが生き残る時代に

# 変化にすばやく適応し、市場で生き残るには？



アジャイル開発で  
実現できること

1

重要だと思っているものから作る

2

迅速かつ継続的にリリースする

3

フィードバックを得る

4

要求の変更をあるべきものとして受け入れ、柔軟に対応する

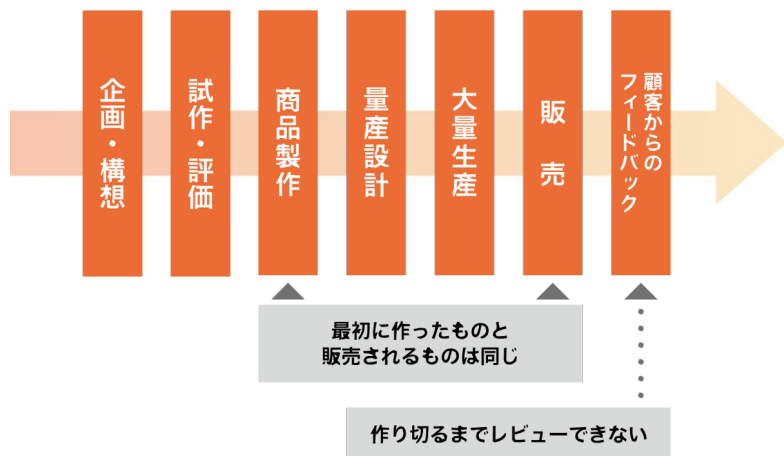
市況の変化に素早く適応することがプロダクトの市場競争力を高め、ビジネスの成功につながる

求められるものを作るためには、重要だと思っているものが世の中に受け入れられるかをテストし、受け入れられなければ変更をしなければいけません。そのためには、迅速なリリースを定期的に行い、顧客の意見を聞き、変更要求があれば対応していくことも重要です。

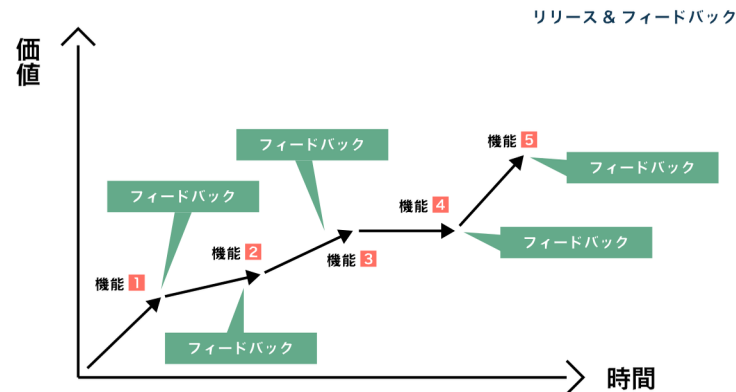
# ウォーターフォール型開発とアジャイル開発の違い

乱暴に言えば、予め決まったものを早く効率よく作りたいならウォーターフォール型開発。  
プロダクトの最終形が決まっていなければアジャイル開発がおすすめです。

## ウォーターフォール型開発



## アジャイル開発



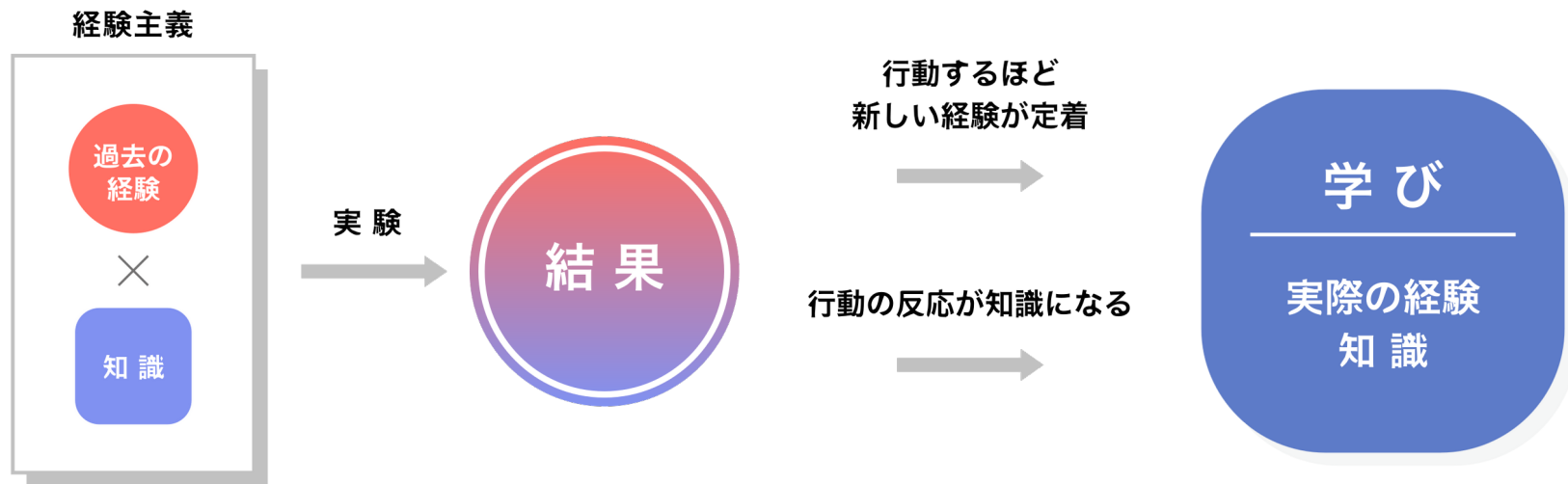
リリース頻度が高いので、  
■ 素早くフィードバックが得られる ■ 要求の変更に柔軟に対応できる

# アジャイル開発をやってみよう！

ここからは、アジャイル開発のフレームワークとして  
近年よく採用されている「スクラム」について説明します

# アジャイル開発のフレームワーク「スクラム」

スクラムは、**わからないことは実践して知識を得よう**という、科学研究で使われる経験的プロセス制御のアプローチを応用したフレームワークです。起源は最も古く 1986 年に発表された概念です。不確実で次々に現れる課題に対しチーム全体でアイデアを出しながら、目標の完遂を目指して障害を乗り越えていくことを目指すために編み出されたものです。



# スクラムチームの役割

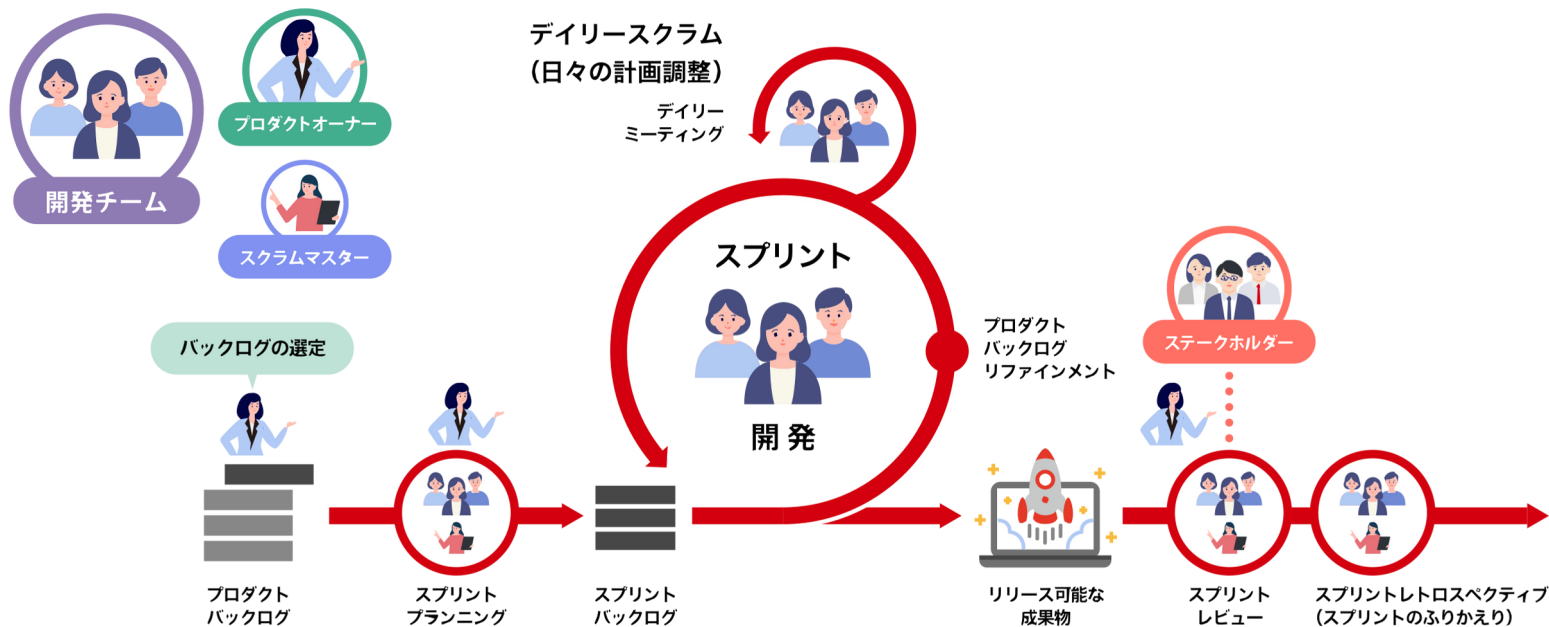
不確実な環境で中央集権的な意思決定を行うと適応が遅れてビジネスチャンスを逃してしまいます。

スクラムでは10名以下の小さなチームが、経験をベースに自律的に意思決定を行い、プロダクトを成長させます。



# スクラムのフレームワーク

スクラムでは、1ヶ月以内の短い期間（スプリント）で、計画・開発・リリース・ふりかえりのサイクルを回していきます。チームは反復的に自分達の仕事の進め方をふりかえり、改善や軌道修正しながら目標に向かって進んでいきます。



# バックログリファインメント — スプリントに入る前の準備 —

## 目的 大雑把な状態のバックログを詳細かし、実現可能な状態にする

誰のための  
イベント？



プロダクトオーナー



開発チーム



ステークホルダー

参加者は？

プロダクトオーナー / 開発チーム  
スクラムマスター / ステークホルダー / その他関係者

### 何をするか

- ☒ 開発チームが PO に質問しながらバックログを共同して書く
- ☒ バックログの記述にはユーザーストーリーの形式が使われることが多い  
「●●というユーザーとして、●●という機能が欲しい。そうすれば●●という利益が得られる。」
- ☒ 具体的なテストも書く

### イベントが終わった時の状態

- ☒ バックログが適切な粒度に詳細化されている
- ☒ 規模が見積もられている
- ☒ パスすべきテストが明らかになっている

スプリントに入る前やスプリント中に行われることが多く、チームはプロダクトバックログの項目を詳細化、優先順位付け、規模感の見積もりなどを行います。開発チームは適切な質問を繰り返してユーザーストーリーや把握しておくべきビジネスルールを理解します。この作業によって、プロダクトバックログが整理され、次のスプリントプランニングに向けての準備が整います。

# スプリントプランニング

## 目的 バックログの着手順を決める、計画に合意する

誰のための  
イベント？



参加者は？

プロダクトオーナー / 開発チーム  
スクラムマスター

### 何をするか

- ☒ 1. バックログの内容について合意する
- ☒ 2. リスク、依存関係、期日を勘案して、着手順を決める
- ☒ 3. 必要に応じて、より小さなタスクに分割する

### イベントが終わった時の状態

- ☒ スプリントゴールが決まっている

スプリントが始まる時に行われる会議です。

チームは「何をやるのか（何を開発するのか）」と「どうやってそれを達成するのか」を決めます。

プロダクトオーナーがプロダクトバックログから選んだアイテム（やるべきこと）をチーム全体で見て、その中からこのスプリントで取り組むべきものを決めます。

次に、開発チームはタスクをより小さな作業に分解します（スプリントバックログ）。この時点で全ての詳細を決める必要はありません。

# デイリースクラム

## 目的

開発チームが自分達の仕事を同期する、誰が何をするのか決める

誰のための  
イベント？



参加者は？

プロダクトオーナー / 開発チーム / スクラムマスター

### 何をするか

- ☒ 短時間 (15 分以内) で終わらせるために、立って、タスクボードを囲んでやる
- ☒ 問題を明らかにするところまでに留め、対処方法については議論しない (検討は別の時間を設けて関係者で集まる)

### イベントが終わった時の状態

- ☒ 全員が何をやればいいのか知っている

デイリースクラムは毎日行われる短い会議で、通常は立ち会いで行います。

「昨日何を達成したか?」「今日は何をする予定か?」「何か問題はあるか?」などを確認し、チームメンバーが他の人が何に取り組んでいるのか、何が進行中なのかを知り、問題があれば早く解決することができます。フレックス勤務や時差などで勤務時間が合わない場合は1日に2回以上実施されることもあります。

## 目的 開発した成果物に対するフィードバックを得る

誰のための  
イベント？



参加者は？

プロダクトオーナー / 開発チーム  
スクラムマスター / ステークホルダー / その他関係者

### 何をするか

- ☒ 1. 予定されていたスプリントゴールについて説明する
- ☒ 2. 開発チームが参加者に成果物をデモンストレーションする
- ☒ 3. 参加者はそれに対して率直にフィードバックする

### イベントが終わった時の状態

- ☒ 成果物に対する学び / 気づきが得られる
- ☒ 必要に応じて、新しいバックログが追加されている

スプリントで開発したものを確認・評価する会議で、開発チームは作成した成果物を披露し、プロダクトオーナーやステークホルダーは、それに対しレビューを行います。レビューで課題が明らかになった場合、チームはその課題に対して必要なアクションを議論し、課題を解消するための具体的なタスクや要件を決定します。それらはプロダクトバックログに新たな項目として追加するか、関連する既存項目に組み込みます。プロダクトオーナーはプロダクトバックログの状態を更新・優先順位を調整し、何が完了したか、何が未完了なのかをチームと共有します。これにより、次のスプリントで何に取り組むべきかを考える基礎を作ります。

# スプリントレトロスペクティブ

**目的** 学びを最大化し、次のスプリントをより良いものにする

誰のための  
イベント？



参加者は？

開発チーム / スクラムマスター

## 何をするか

- ☒ 1. ふりかえりの目的を決める（業務プロセスを改善 or チームの問題を改善？）
- ☒ 2. データを収集する
- ☒ 3. 分析・アイデアを出す
- ☒ 4. 何をすべきかを決定する

## イベントが終わった時の状態

- ☒ うまくいく / うまくいかないパターンが発見されている
- ☒ 探索すべき事象と探索する方法が明らかになっている

スプリントの最後に行う会議で、「どこがうまくいったか」「何がうまくいかなかったか」を洗い出し、「どうすればもっとよくなるか」を話し合います。これにより次回のスプリントに向けての改善点を見つけ、目標に近づく方法を探ります。

この会議はチームの成長と改善を促します。具体的な改善策を提案し、それを次のスプリントで試すこととなります。これにより、チームは絶えず自己改善と学習を続けることができます。

# スクラムの基本原則と価値基準

スクラムの基本原則と価値基準は、プロジェクトの進行と改善のための重要な要素です。スクラムチームの全員が持つべき思考と行動の原則で、プロジェクトを成功させるための基盤となります。これらを共有することで、チーム内の信頼関係が築かれて協力的な作業が促進されたり、チーム内の情報共有によって問題解決がスムーズに行えるようになります。

## 3つの基本原則

### 透明性

プロジェクトの進行状況、課題、成果物などをチーム全員や関係者に明確に共有する。  
全員が同じ情報を持っているため、誤解や混乱を減らし、効率的に取り組むことができる。

### 検査

プロジェクトの進行状況や成果物を定期的にチェックし、評価する。  
課題や偏りを早期に発見し、修正するための基盤となる。

### 適応

検査で得た情報に基づき、プロジェクトの方向性や仕事の進め方を必要に応じて変更する。  
変化するビジネス環境やプロジェクト内の課題に柔軟に対応するための手段。

## 5つの基本原則

### 確約

チームはゴールを達成するためお互いにサポートし合うことを確約する

### 集中

チームはゴールに向けて可能な限り進歩できるよう作業に集中する

### 公開

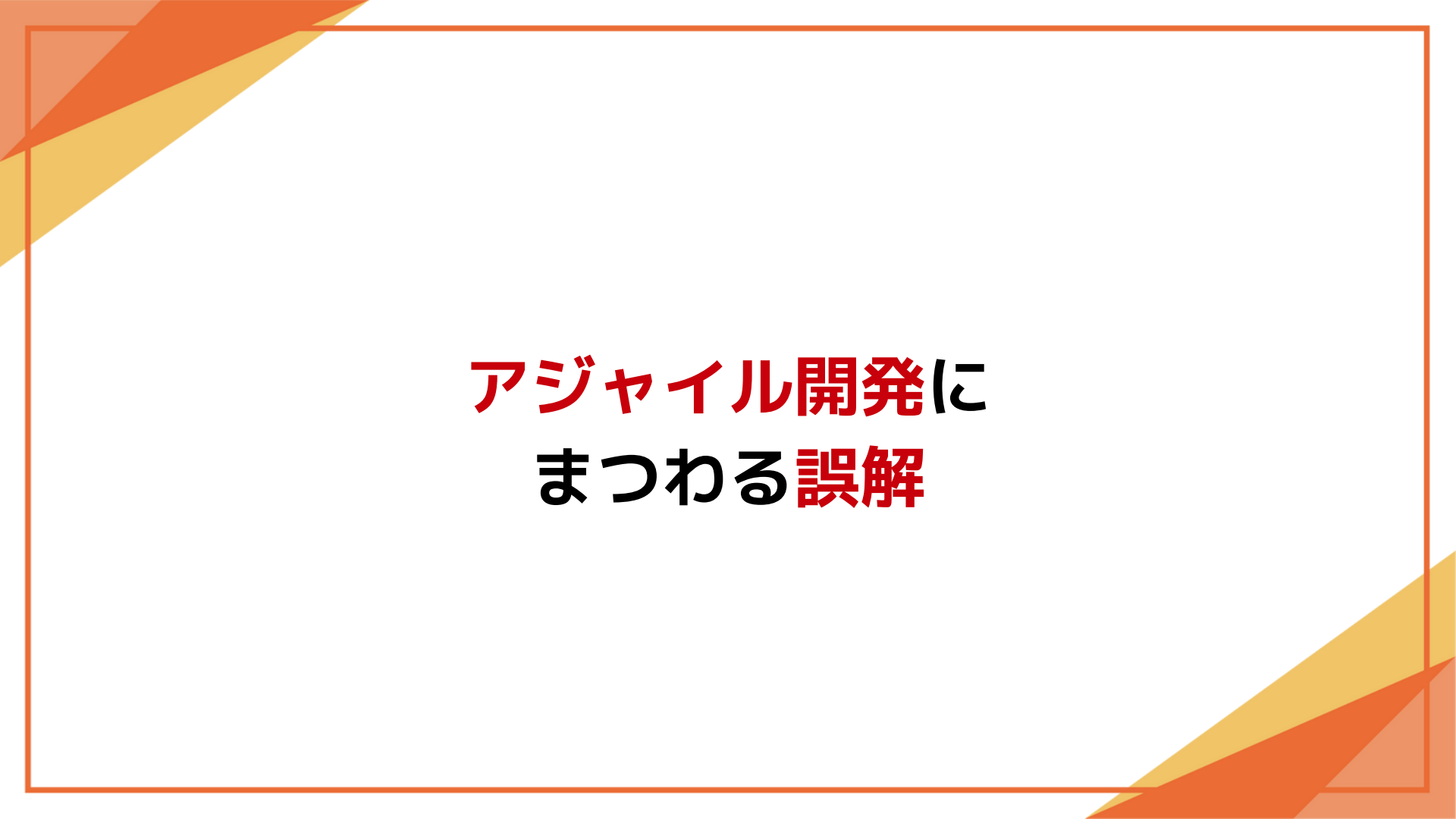
チームとステークホルダーは作業や課題を公開する

### 尊敬

メンバーは互いに能力のある独立した個人として尊敬し、尊敬される

### 勇気

メンバーは正しいことをする勇気や困難な問題に取り組む勇気を持つ



# アジャイル開発に まつわる誤解

## ▶ 計画通りに進みません。どうすればいいですか？



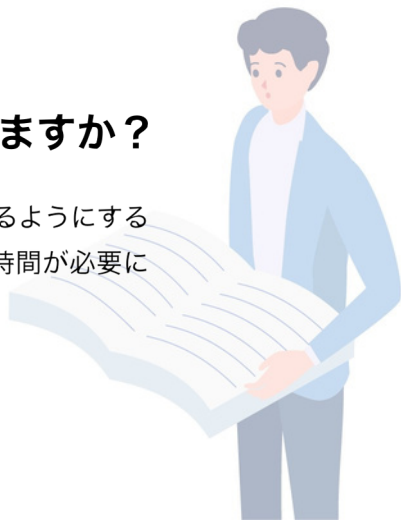
複雑な問題解決を目的とするアジャイル開発では、むしろ計画が常に変わることを前提にしています。計画が通りに進まないことは、チャンスと捉えることもできます。それは新たな学習の機会であり、仕事のやり方や製品を改善するための新たな視点を提供してくれるかもしれません。スプリントレトロスペクティブで、何がうまくいかなかったのか、どうすれば改善できるのかを考え、同じ問題が繰り返されるのを防ぎましょう。大切なのは、計画を守ることも経験によって得られた反応を受け入れ、変化に適応していくことを優先することです。

## ▶ トップダウンで形からスクラムを導入して、アジャイルなチームになりますか？



スクラムを成功させるためには、チームメンバーや関係者がスクラムの原則と価値を理解し、実践できるようにする必要があります。真の変革を達成するためには、組織全体でのコミットメント、教育、支援、そして時間が必要になるでしょう。

社内に推進者が少ないと感じたら、コミュニティを活用し、同志や相談先を増やすのもおすすめです。



## —— 参考資料 ——

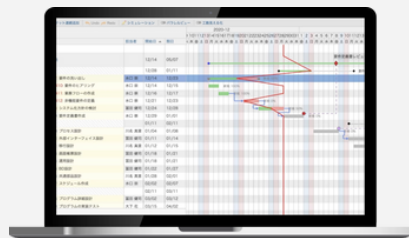
- アジャイルソフトウェア開発宣言
- スクラムガイド（2020年11月版）

プロジェクト管理をより本格的に行いたい企業様は、ぜひ、Lychee Redmineをご利用くださいませ。他のお役立ち使用も貴社のプロジェクト管理に貢献できれば幸いです。



最後までご覧いただき、ありがとうございました！

Lychee Redmine  
30日間無料お試し



無料体験版を  
今すぐ試す

お役立ち資料



お役立ち資料は  
こちら

もっと詳しく



Webサイトを  
見る



株式会社アジャイルウェア  Agileware

アジャイルウェアホームページ : <https://agileware.jp/>

LycheeRedmine導入のご相談 : <https://lychee-redmine.jp/contact/>

お問い合わせ先 : <https://agileware.jp/contact/other-form/>