

ソフトウェア・シンポジウム 2020

プロダクトマネージャーが求める アジャイル開発におけるソフトウェアテストのあり方 と生産性向上への取り組み

株式会社リクルートライフスタイル
データソリューションユニット マネージャー
坂東壘

株式会社ProVision
羽鳥温子（共著者）

Agenda

- 背景・これまでの取り組み
- 残り続けた課題
- 目指すべき方向
- 取り組みと結果
- まとめ

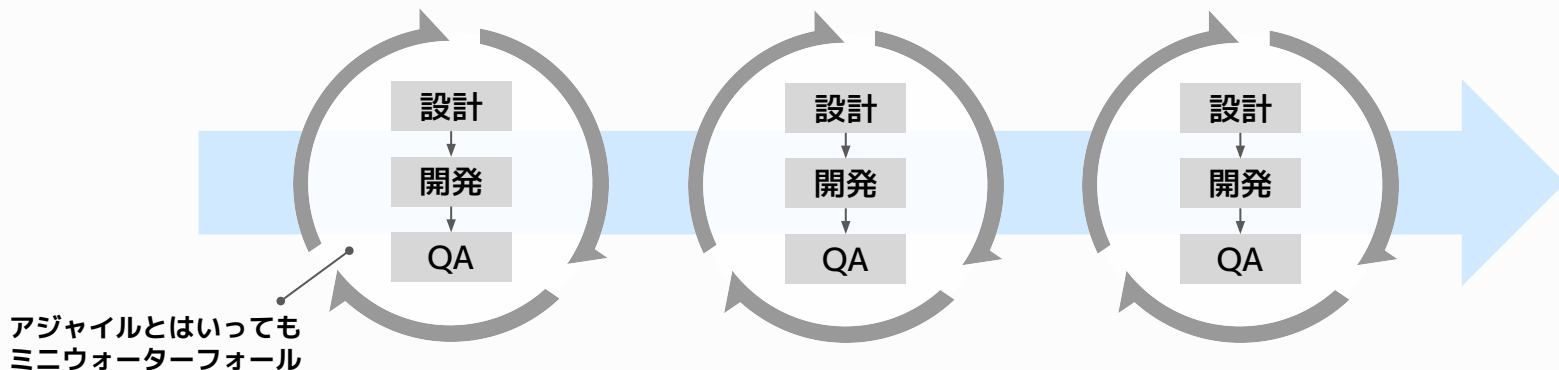
Agenda

- 背景・これまでの取り組み
- 残り続けた課題
- 目指すべき方向
- 取り組みと結果
- まとめ

アジャイルチャレンジと生まれた課題

“じゃらんnet”の開発の一部をウォーターフォール開発からアジャイル開発に変更し、生産性の向上に向けてチャレンジを続けてきた。しかし、生産性向上に向けてアジャイルを意識しただけでは、ミニウォーターフォールという傾向が見られ、開発メンバーやQAメンバーの役割が不変であることも多く、結果として生産性向上と逆効果となる可能性もあった。

■役割分担によるスタイル例

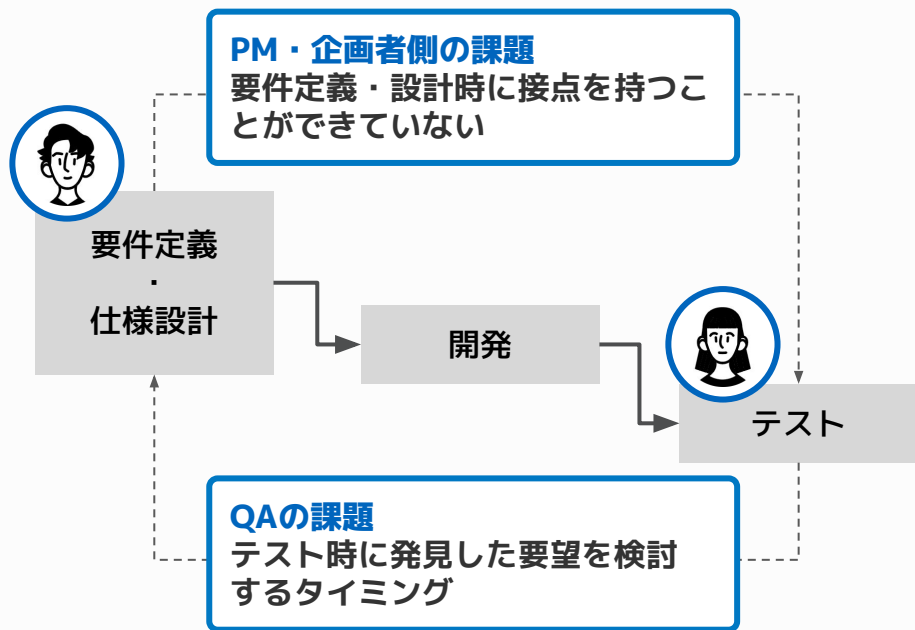


新たな変化として“QA”の役割の変化に取り組んだ

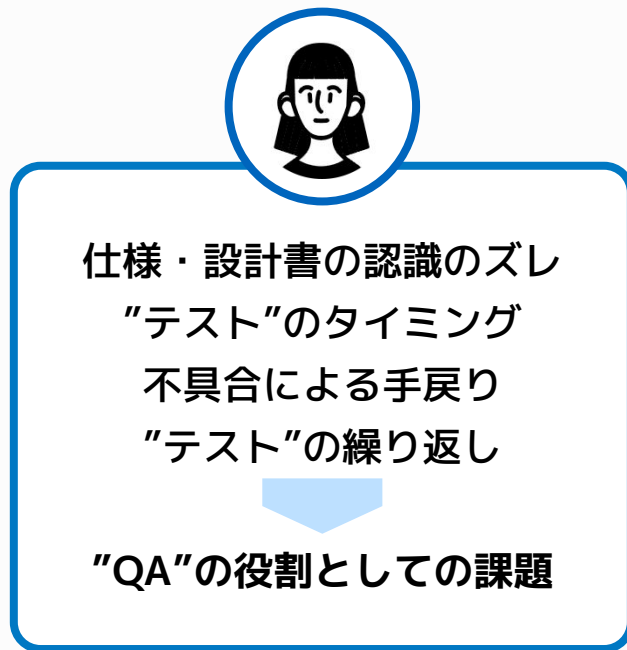
※企画者の認識の変化、リリースタイミングや開発の規模の変化のみ

自組織における“ソフトウェア・テスト”の課題

タイムボックス的な開発 における接点の壁



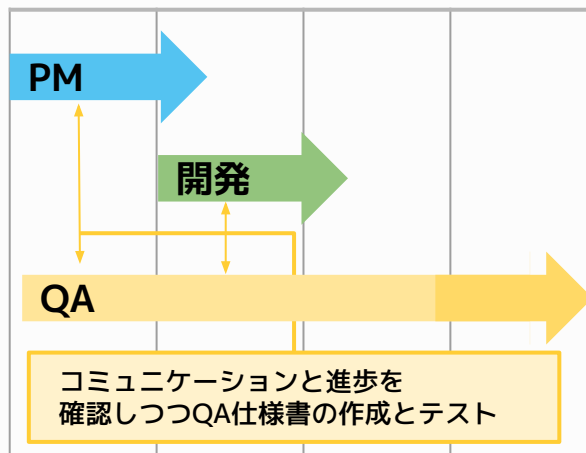
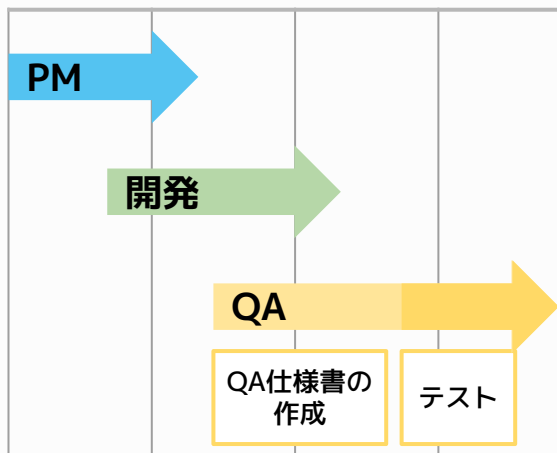
ソフトウェアの“テスト”実施の タイミングによる負の存在



アジャイル開発における「QA/テスト」の立ち位置

企画者・開発者との緩やかな連携による改善

固定化されたタイムボックスの中で動くのではなく、PM（企画者）や開発者と緩やかな連携を図る。



緩やかな連携

開発はタイムボックスで開発を実施し、QAはタイムボックスに沿わない

- ❑ アジャイルな意識を持って、ミニウォーターフォールであることも多いため、各役割のタイムラインは変化せず、QAが関わるタイミングもウォーターフォールの時と同様
- ❑ 明確なテスト期間を設けずに開発の進捗に応じたテスト設計・テストを実施。
- ❑ 常に各役割のメンバーの状況を把握することで、柔軟にタスクに取り組めるようにする。

“QAスキーム”について

“QAスキーム”フロー	QAタスク詳細
企画フェーズ (仕様書や設計書へのFB)	<ul style="list-style-type: none">❑ 企画会議に参加する（案件を知る）❑ スケジュールの話し合いにも参加（他案件との兼ね合いも把握できる）
開発フェーズ	<ul style="list-style-type: none">❑ 開発が終わる前に、企画フェーズで整理した内容をもとに項目書設計❑ 企画者向けの項目書確認
QAフェーズ	<ul style="list-style-type: none">❑ テストの実施❑ バグの改修確認❑ 試験完了

■生産性向上意識した“QAスキーム”アプローチ

- 開発メンバーはタイムボックスに沿って開発を実施するものの、QAメンバーはタイムボックスに沿わないことで、生産性向上アプローチにチャレンジ

“QAスキーム”における役割の変化によりQAメンバーの意識改善へ
シフトレフトによるテストフェーズの効率化と企画・仕様の手戻り防止を目指す

生産性向上アプローチへのQAの役割変化

	Before	After
仕様・要件	<input type="checkbox"/> 仕様・要件はインプットがメイン	✓ 仕様に対して、ユーザ目線・機能目線で指摘
試験設計	<input type="checkbox"/> 決まった仕様の範囲内で設計	✓ 企画会議などで把握した内容を考慮し、設計
バグ・不具合	<input type="checkbox"/> 機能不具合のみ	✓ 機能不具合に加えて改善提案も

QA・テストという観点から少し離れた上流工程（仕様検討など）の役割も担うことで、プロダクトの仕様に早い段階から精通し、プロダクトの品質（製品品質だけでなく利用時品質も）を意識するようになる

プロジェクトにおける“テスト”だけという役割で、プロジェクトへ関われきれていない想いもあったが
“QA”の役割変化により、案件により携われるようになったことで想い入れも変化
QAメンバーの「サービス」に対する「品質意識」は高まった

“生産性改善アプローチ”において変化しなかったこと



QA
の取り組み

QAメンバーの役割見直しによるプロダクトに対する意識の変化

- QAの役割変化によって品質への意識が変化



生産性改善
に向けた買

チームのアウトプットとしてPMが期待した生産性改善が感じられない

- QAメンバーのシフトレフトを中心とした取り組みで、企画の品質への意識は変化したものの生産性改善にはそこまで繋がらない
- 企画部分ではなく、実装・テストの部分でのQAメンバーの役割が変化していない



期待値通りの
結果との差分

生産性改善に向けて課題を改めて見直す

- 改めて、QAスキームと合わせてチームとして課題を見直す

生産性の改善の
本当の課題は
どこにあるのか？

Agenda

- 背景・これまでの取り組み
- **残り続けた課題**
- 目指すべき方向
- 取り組みと結果
- まとめ

“QAスキーム”に残り続ける課題

課題	課題における原因
不具合の増加による生産性が未改善	<p>生産性を期待・意識するあまり、QAテストへ回されるプロダクト品質の低下によって不具合の増加に繋がった</p> <ul style="list-style-type: none">❑ バグが多いことで、手戻り・バグ再確認の発生増加❑ 初歩的・要件自体を満たさないバグの増加❑ バグ発生増に伴い企画者との擦り合せ時間の増加
プロダクト品質の意識のズレ	<p>アジャイル開発に変化したことで、プロダクトの開発フローにおける品質への意識に変化が生まれた可能性</p> <ul style="list-style-type: none">❑ ウォーターフォール開発=“厳しい”・アジャイル開発=“緩い”という意識❑ QAメンバーのテストで品質を担保という意識❑ メンバー間（開発⇔QA間）での「品質担保」の意識のズレ



“QAスキーム”の課題をチーム全体で見直す必要性

QA内での改善努力だけで終わらせて解決できるものは少なく、“QA”スキーム課題はチームにおける課題と捉える

Agenda

- 背景・これまでの取り組み
- 残り続けた課題
- **目指すべき方向**
- 取り組みと結果
- まとめ

“QAスキーム”における課題に対して目指すもの

改めて、“QAスキーム”と生産性の関係性を見直す

チームとして、プロダクトの生産性改善を目指すために
“QAスキーム”における「品質」と「テスト」の違いを整理する

違いを整理した上で“QAスキーム”の再定義へ

- “QAスキーム”を改めて見直し、実装フェーズ以降のQAとテストの関係を改善
 - これまで“シフトレフト”によるQAメンバーの上流工程への染み出しという意識だけで、実装フェーズ以降の“QAスキーム”の見直しが置き去りにされていた
- QAメンバーだけが“テスト”だけをやる体制は正しいのか
 - 「品質担保」はテストフェーズでの“テスト”によって、QAメンバーが担保するという意識が強かった
 - 結果として、品質担保のための“テスト”はQAメンバーがテストフェーズで実施という考えを見直す必要があった
- “品質”と“テスト”の違いを明確にし、QAメンバーの役割を再定義する
 - 「品質」とは、QAがリードして、チーム全体で担保するものであり、品質担保のための「テスト」もQAメンバーだけで担保するものではなく、「テスト」自体は誰が実施してもよいものである

Agenda

- 背景・これまでの取り組み
- 残り続けた課題
- 目指すべき方向
- **取り組みと結果**
- まとめ

QAスキーム（役割）の変化

Before

1. 項目書作成
2. 企画者による試験設計書確認
3. QAテスト

テストフェーズにおいて
QAのみがテスト

After

1. 項目書作成
2. 企画者による試験設計書確認
3. 開発メンバーが項目書をもとに開発テスト
4. 開発メンバーの気づきが項目書に反映されることで項目書がブラッシュアップ
5. QA試験項目書の見直し
6. QAテスト
※開発テストで同項目がテストされていることにより、バグの減少

テストフェーズに関わらず、開発メンバーも
QA項目におけるテストを実施

品質をQAの
「テスト」
以外でも担保

品質担保へのチームの
意識変化

“QAスキーム”の中にQAメンバーだけでなく、チームとしてに加わることにより
結果としてQA課題の改善へと繋がった

QAスキーム（役割）変化によって生まれた結果

生まれた結果

品質をQAの「テスト」以外でも担保

品質担保へのチームの意識変化

結果による改善

- ❑ 開発メンバーがQAが作成した項目書でテストを実施することで以下が担保される
 - ❑ 項目書の抜け漏れの担保
 - ❑ テクノロジー観点でのレビュー効果
 - ❑ テクノロジー面レビュー・共有によるQAの成長（新しいテクノロジーに適應できないことで、QAテストの正確性が担保されていないことに繋がる可能性の減少）
 - ❑ 項目書ベースでの実装漏れの確認・担保
 - ❑ 開発メンバーのタイミングで不具合が確認されることで、QAの「テスト」タイミングでの手戻りの減少
- ❑ 開発テストをはじめてから、開発側からもっとこうしたらいいのでは？のような連絡が増えた（今までだとQAの「テスト」に関心が不明瞭であった）
- ❑ KPTなどの振り返りでも全員が開発テストをやってみたいという前向きな意見の増加
- ❑ 開発メンバーとQAとのやりとりも増えた

QAサイロ化の脱却による、チームとして「品質担保」へ

QAがテストでのみ「品質担保」するスキーム（意識）からチームとして「品質担保」するスキームへ

“QAスキーム”変化がもたらしたゴール

プロダクト品質の向上から生産性改善へ

旧来のQAの「テスト」による「品質担保」から
チームとしての「品質担保」による品質改善により
生産性最大化

生産性最大化に向けて

- **昨対比較による効果実感について**
 - 2018年度の不具合検出率「平均6.0%」 / 2019年度の不具合検出率「平均2.3%」
 - QAテストによる不具合検出率の減少による手戻りの減少
- **開発テスト工数増を上回る効果**
 - 開発テスト時の事前における手戻り減少に加え、項目ブラッシュアップにおける不具合の見逃しリスクの軽減によるメリット
 - テクノロジーロックインなどQAサイロ化の脱却の一端を担う

Agenda

- 背景・これまでの取り組み
- 残り続けた課題
- 目指すべき方向
- 取り組みと結果
- **まとめ**

チームにおける生産性向上までの道のり



アジャイル開発におけるQAの立ち位置の変化（緩やかな連携）

- 開発企画段階から入る試みを我々も実施しており、その中で一定のQAのモチベーションやメリットも得ていたため、チームとしての手応えは感じていた
- しかし、もともと早いタイミングからQAが関わればよいという話（シフトレフト）もあるが、開発後・QA後のタイミングでの「品質担保」については限界があった



サイロ化されたQA意識からの脱却

- QA = 「テスト」ではなく、「品質担保」の番人としての役割を意識
- チームとして「品質担保」という意識を持った中で新たな役割を担う



新たなソフトウェア・テストのあり方を模索するため“QAスキーム”の刷新にチャレンジ

- チームとして新たな“QAスキーム”への取り組みが生産性改善に繋がった

アプローチ結果としての変化

生産性の改善は、品質担保を軸とした“QAスキーム”の変化から生まれた

アジャイル開発における生産性最大化は
“QAスキーム”のあり方を刷新することで生まれた
品質担保なくしては実現されない

「品質担保」はチームの意識によって実現される

- QAメンバーのシフトレフトだけでは生産性改善も部分的な改善に留まる
 - QAメンバーの役割を上流工程に染み出しただけでは、一部のプロセスが改善されたが、プロダクトの生産性という点では大きな改善が見れなかった
- チーム全体で役割を制限せずに、「品質担保」に取り組むことが重要
 - 「品質」というと、QAメンバーの「テスト」で担保という意識がどうしてもメンバーの中にあるが、「品質」をチーム全体で意識することで、結果として高いレベルで「品質担保」を実現できることに至った
- チームに適したソフトウェア・テストの発見による生産性向上の実現
 - プロセスや役割の変化によってプロダクトへの取り組みの改善は見られるかもしれないが、「品質担保」を前提とした“QAスキーム”の改善により生産性向上に繋がった。