

4. 実践チーム開発

4.1 アプリケーションのアイデアを決める

4.1.1 オズボーンのチェックリスト

1つのテーマを元に、「転用」「応用」「変更」
「拡大」「縮小」「代用」「再利用」「逆転」「結合」
の9項目を使ってアイデアを広げる

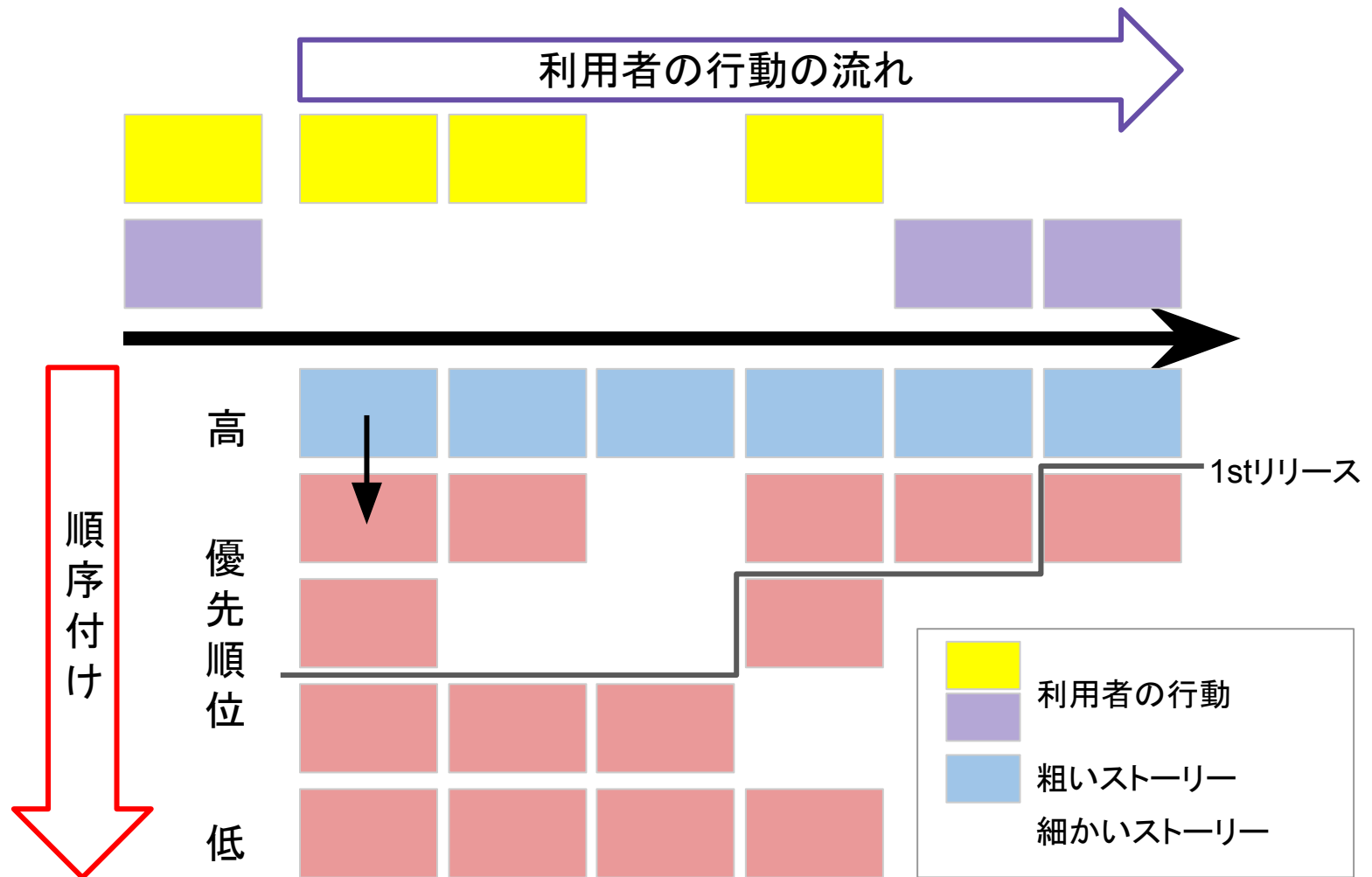
転用: 他の使い道を探す	応用: 類似する物を探す	変更: 変えてみる
拡大: 拡大してみる	縮小: 縮小してみる	代用: 置き換えてみる
再利用: 配置や並びを替えて みる	逆転: 逆にしてみる	結合: 組み合わせてみる

4.1.2 リーンキャンバスで プロダクトの特徴を明確にする

<p>なぜお客様は喜んでくれる？ 課題</p> <p>1</p> <p>既存の代替品</p>	<p>具体的に 何を提供する？ ソリューション</p> <p>4</p>	<p>なぜお客様は我々から買いたくなる？ 独自の価値提案</p> <p>3</p> <p>わかりやすい コンセプト</p>	<p>なぜ我々は喜ばせることができる？ 圧倒的優位性</p> <p>9</p>	<p>誰に喜んでほしい？ 顧客セグメント</p> <p>2</p> <p>アーリーアダプター</p>
<p>コストはどれぐらいかかる？ コスト構造</p> <p>7</p>	<p>喜んだことを どうやって知る？ 主要指標</p> <p>8</p>	<p>どうやってコミュニケーションする？ チャネル</p> <p>5</p>	<p>結果として我々には何がもたらされる？ 収入の流れ</p> <p>6</p>	

4.2 アプリケーションの内容を決める

4.2.1 ユーザーストーリーマッピングで 製品の機能を洗い出し、リリース計画を立てる



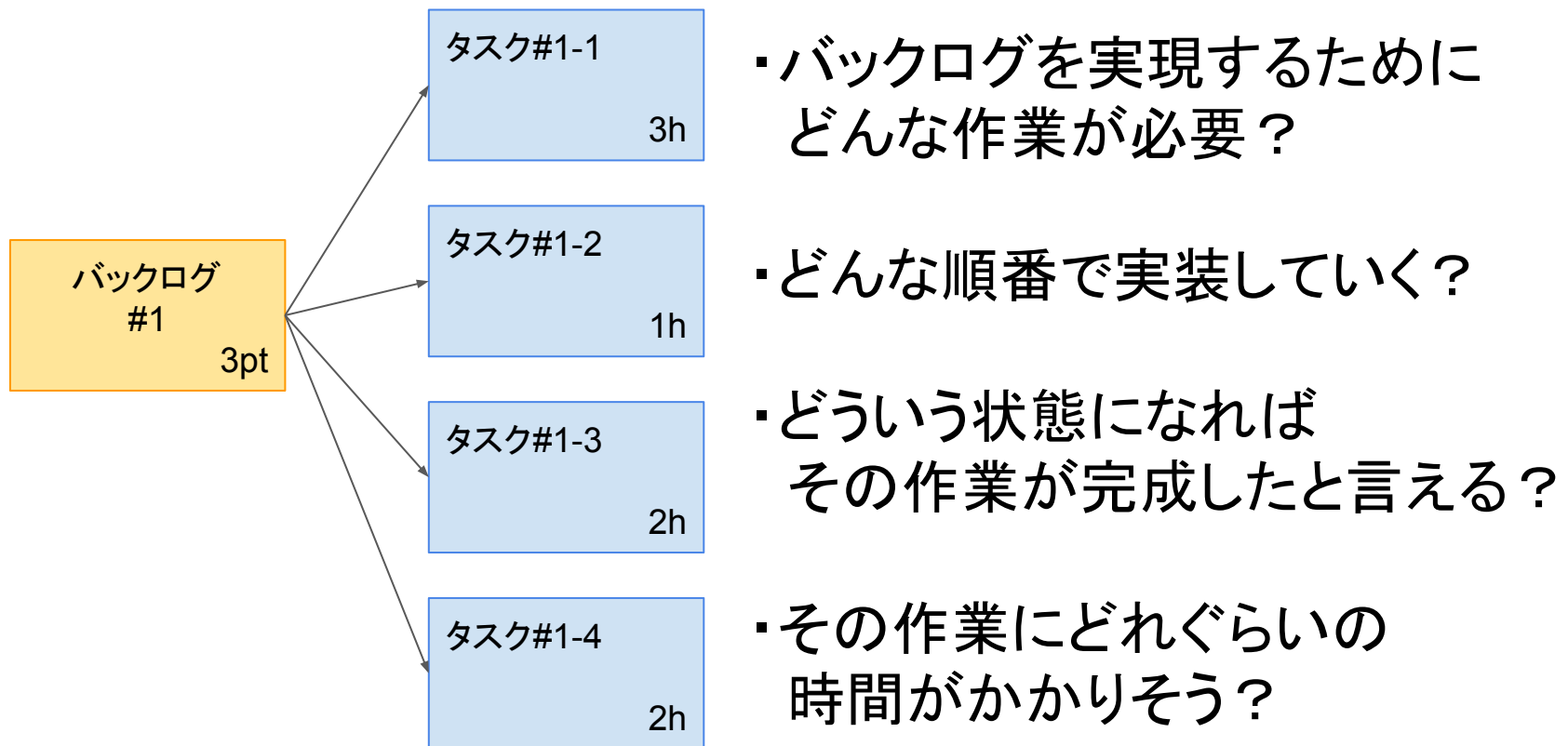
4.2.2 1stリリースに必要なバックログに 優先順位をつけ、プランニングポーカー で見積もる



- みんなで見積もる
- 認識を合わせる
- 時間をかけ過ぎない
- 大きな数字の
小さな誤差は気にしない

4.3 スプリントの計画を立てる

4.3.1 バックログを実現するために必要なタスクを洗い出し、作業工数を見積る



4.3.2 カンバンを作る

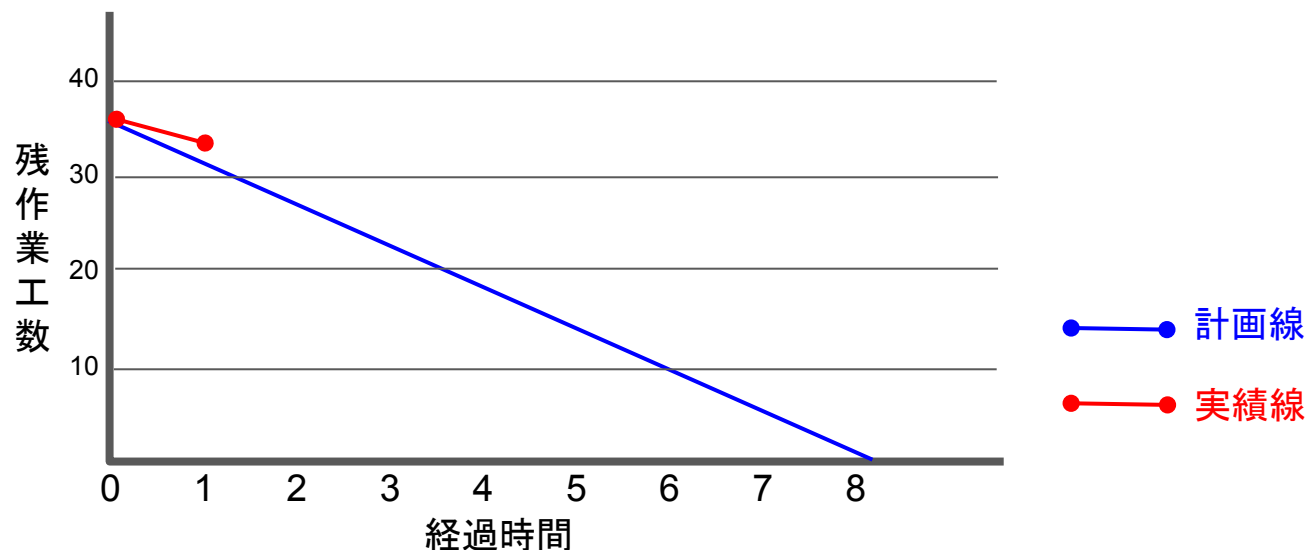
バックログ	To Do	Doing	Done
<div>バックログ #1 3pt</div>	<div>タスク#1-1 3h</div> <div>タスク#1-2 1h</div> <div>タスク#1-3 2h</div> <div>タスク#1-4 2h</div>	<div></div>	<div></div>
<div>バックログ #2 2pt</div>	<div>タスク#2-1 1h</div> <div>タスク#2-2 1h</div> <div>タスク#2-3 3h</div>		

4.3.3 バーンダウンチャートを作る

縦軸: 残作業工数(時間)

横軸: 時間

で作ってみましょう



開発が始まったら、

60分毎に作業中のタスクの残作業時間を見直し、
バージョンダウンチャートの実績を更新しましょう

4.4 チーム開発を経験する ～スプリント#1～

【実践すべきプラクティス】

- ・テストコード(テスト自動化)
- ・ペアプログラミング
 - ・チーム内でコマ毎にローテーションを行う
- ・コマ毎にカンバンを更新する
- ・リファクタリング
- ・GitHub Flowの流れでレビューを実施する