

投稿日 2024/03/29

プロダクト機能の見積

見積に求められること

プロダクトに関する要求を整理した後は、開発にどのくらいのコスト・期間が必要なのかを見積らなければなりません。

ここでの見積もりの目的は以下の2点です。

1. 開発にどのくらいの期間や予算、体制が必要になるのかを素早く知ること
2. コスト超過・期間超過を含むリスクを早期に検知し、その対策を計画に組み込むことでプロジェクトの成功可能性を高めること

この目的を果たすために重要なのは以下の点になります。

- プロダクト価値をもたらすユーザーストーリーごとに見積もること
- スピーディに見積もること
- リスクを定量化すること
- 優先順を明確にすること

なお、本ページではプロダクト機能開発の見積もりにフォーカスし、インフラ費用やSaaS費用などは対象外とします。

これらの費用については別途見積もりを行ってください。

ユーザーストーリーごとに見積もる

開発見積のインプットは、ユーザストーリーマップとして整理された要求です。

そしてそのアウトプットは、ユーザーストーリーごとの見積になります。

ユーザーストーリーが前提となる理由は、「プロダクト価値」の優先度順に開発を進められること、また、開発の優先度を変更する際にもその影響を最小限に抑えることができるためです。

スピーディに見積もる

新規事業開発においては、最初から要件や解決方法がしっかりと固まっているわけではありません。開発中にもさまざまなことを学習しそれを事業・プロダクトに取り入れていく新規事業開発では、提供したい価値や機能も変わっていきます。このような変化に迅速に適応するためには、見積もりにもスピード感が求められます。

リスクを定量化する

新規事業開発では、実現手段が明確でない等、さまざまなリスクが存在します。そういった中でも「いつまでにできるのか」「どれだけのコストが必要になりそうなのか」という質問には回答せざるを得ません。

このためには、どのユーザーストーリーがリスクを内在しているのか、そのリスクの大きさはどの程度なのかを定量的に評価し、そのための戦略を考えていく必要があります。

優先度を決める

見積もりをした結果、予算や想定していた開発期間を超えてしまいそうな場合は、スコープの調整が必要になります。

スコープを決めるためには優先度が必要ですが、優先度づけは難易度の高い課題です。

この優先度づけに関して、我々がおすすめするのはRICEというフレームワークです。

- [RICE: Simple prioritization for product managers](#)

これは、「RICE」を構成する文字で表現される以下の4つの指標によって、ある種の費用対効果を計算し、優先度を測る方法です。

- R(Reach): 影響を与えられる人数
- I(Impact): 個々人に与えられる影響の度合い
- C(Confidence): RやI、Eの確度
- E(Effort): 当該機能の開発に必要な工数

優先度を測るスコアは以下の式で算出します。

$$\frac{\text{Reach} \times \text{Impact} \times \text{Confidence}}{\text{Effort}}$$

RICE

事業オーナーとともにこの値を算出することで、優先度を適切に判断しやすくなるでしょう。

見積の進め方

見積時点では、実現する機能の詳細まで固まっていないことも多いです。このようなケースでは、細分化したタスクの工数を積み上げていく見積手法ではなく、ユーザーストーリーごとにストーリーポイントをつけていく見積手法の方が適しています。

ストーリーポイントはユーザーストーリーの作業の大きさ・開発内容の複雑さや内在するリスクなどを一体として表す単位です。ここで重要なのは他の作業との相対値で、2ポイントのストーリーは、1ポイントのストーリーの2倍の規模であることを示しています。

この手法は、一般に絶対的な見積もりよりも相対的な見積もりの方がうまくこなせる¹我々には適した方法です。

ユーザーストーリーごとにポイントを付与した後、1ポイントあたりがどのくらいの開発工数に相当するのかをチーム内で議論し、

「ポイント×1ポイントあたりの開発工数」という計算にて全体の開発工数に変換します。

(なお、開発が進めば「1スプリントでどの程度のポイントを消化できるか」というベロシティがわかるようになり、それに応じて開発ペースを調整できます)

ストーリーポイントの見積

ストーリーポイントを付与する方法としては、Tシャツサイズ見積、またはプランニングポーカーが有名です。

それぞれの考え方は以下の記事で詳しく説明されているのでご参照ください。我々のお勧めは「プランニングポーカー」です。

- [Tシャツのサイズは意外と便利？プロジェクトのTシャツ見積もり](#)
- [プランニングポーカー: アジャイルな見積もりのためのオールイン戦略](#)

どちらの方法を取るにせよ、見積もり際には以下のポイントを押さえておく必要があります。

- ポイントの共通認識を持つ
- リスクを見積もる
- チームで見積もる

ポイントの共通認識を持つ

ストーリーポイントは相対的な値となるため、その基準についてはチーム内で共通認識を持つ必要があります。例えば、DB上のデータを単に画面に表示するだけのユーザーストーリーは2ポイントとする、というような基準を作ります。そうすれば、後のユーザーストーリーは「それより3倍難しい」「2/3程度だ」というように、見積を素早く進めることができます。

なお、割り当てるポイントは、フィボナッチ数列(1、2、3、5、8、…)のように、差がどんどん大きくなる数列を利用しましょう。

人の見積もりは、規模が大きくなるほど不正確になるためです。例えば見積もりにおいて1ポイントと2ポイントの間には明確な差がありますが、80ポイントと81ポイントという見積もりに大きな差はありません。

リスクを見積もる

リスクの見積については、[多点見積りとスケジューリングの実践](#)を参照ください。

我々もこのアプローチを採り、以下のようにして、個々のユーザーストーリーのリスクを計算しています。

1. ユーザーストーリー毎に、およそこのくらいであろうという「平均的なストーリーポイント」だけでなく、リスクが顕在化した時の「最悪の場合のストーリーポイント」をチームで決める
2. ユーザーストーリーのリスク値として、「最悪の場合のストーリーポイント」-「平均的なストーリーポイント」を算出する

こうして個々のユーザーストーリーのリスク値が算出できます。リスク値を「リスクバッファ」という期間に変換する際は、全ユーザーストーリーのリスク値の2乗和平方根を取りましょう。

チームで見積もる

見積もりはチームで行うことが重要です。個人の見積もりでは、個人の経験や知識に依存してしまうため、チームでの見積もりよりも不正確になります。また、チームで見積もることで、チーム全体の知識が共有され、チーム全体での見積もりの精度が上がります。

工程全体の見積

ユーザーストーリーの見積では、ユーザーストーリー単位の設計や実装・単体テストの見積が可能です。一方で、ユーザーストーリー横断で考えるべき方式設計や、初期プロダクトリリース時の結合テスト・システムテスト等については含めることが困難です。

それらについては、工程別の工数割合からそれらのフェーズの工数を算出することを推奨します。
用いたデータはIPAが公開している以下の分析データです。分析対象プロジェクトのプロファイルは新規事業開発のプロファイルと一致するものではありませんが、現時点で最も頼り得るデータとして使用しています。

図表 7-1-3 ● 工程別の実績月数の比率の基本統計量（新規開発）

[比率]								
工程	N	最小	P25	中央	P75	最大	平均	標準偏差
基本設計	193	0.046	0.159	0.198	0.247	0.542	0.207	0.078
詳細設計	193	0.026	0.133	0.175	0.206	0.454	0.175	0.066
製作	193	0.063	0.193	0.233	0.281	0.598	0.249	0.088
結合テスト	193	0.038	0.141	0.184	0.233	0.476	0.193	0.074
総合テスト	193	0.020	0.123	0.176	0.219	0.557	0.176	0.080

[ソフトウェア開発データ白書2018-2019](#) P.177より引用(2022年1月18日閲覧)

こちらの表の中央値をベースにして、各工程にかかる時間を算出します。

工程	中央値	「実装・テスト」を100%としたときの割合
外部設計	0.198	48.5%
実装・テスト ²	0.408	100.0%
結合テスト	0.184	45.1%
総合テスト	0.176	43.1%

上記の表を元に、「実装・テスト」の工数から、外部設計・結合テスト・総合テストのおおよその工数を割り出します。


以上の手法にて、新規事業開発におけるアプリ見積もりをスピーディに進めることができます。


1. Jonathan Rasmusson、[『アジャイルサムライ—達人開発者への道』](#)、オーム社、2011。[🔗](#)
2. ソフトウェア開発データ白書では「内部設計」「製作」なっていますが、我々の開発実態を踏まえて言い換えています。
[🔗](#)



/* Recommend */

「新規事業開発」 のおすすめ記事はこちら

この記事に関連する記事もお読みください。

 新規事業開発
サービス開発の進め方
2024/03/29  4

 新規事業開発
プロダクトの言語化
2024/03/29  3

 新規事業開発
仮説検証とMVPへの向き合い方
2024/03/29  2

最近投稿された記事も用意しました。

 新規事業開発
サービス開発の進め方
2024/03/29  4

 新規事業開発
プロダクト機能の見積
2024/03/29  2

 新規事業開発
プロダクトの言語化
2024/03/29  3

「新規事業開発」 で最も読まれている記事を以下にまとめています。

 新規事業開発
新規事業開発のステージ・ゲート
プロセスとその評価基準
2024/03/29  21

 新規事業開発
新規事業スタートガイド
2024/03/29  48

 新規事業開発
仮説検証とMVPへの向き合い方
2024/03/29  2