

[ホーム](#) [タイムライン](#) [トレンド](#) [質問](#) [公式イベント](#) [公式コラム](#) [キャリア・転職](#) **NEW** [Orga](#)

i この記事は最終更新日から1年以上が経過しています。

📌 Engineering Manager Advent Calendar 2019 1日目

 @hirokidaichi (広木 大地) in  株式会社Rector

エンジニアリングマネージャ/プロダクトマネージャのための知識体系と読書ガイド

アーキテクチャ マネジメント #エンジニアリング

最終更新日 2021年08月02日 投稿日 2019年12月01日

本記事は、Engineering Manager Advent Calenderの 1 日目です。

はじめに

エンジニアリングマネージャ(EM)と呼ばれる職務を設置する企業が増えてきました。

私たちの主催したイベントEOF2019でも700名近い方に参加していただき、また多くの方にご協力いただき成功裏に終わることができました。

EM Meetup/EM.FMなどのムーブメントの中心の一翼を担わせていただき、その高まりを感じる一方で不安も感じます。このエンジニアリングマネージャという職務は非常に多岐にわたるケースが存在していますし、必要だとされるスキルもまちまちです。そして、多くの場合、その企業のステージや状況ごとに求めるものは違います。また、求めていることを明文化することすらされていないケースも存在します。

このことから、エンジニアリングマネジメント自体が一時的な潮流として消費され、消えていってしまうのではと危惧しています。このようなバズワードとしての消費は、



2154



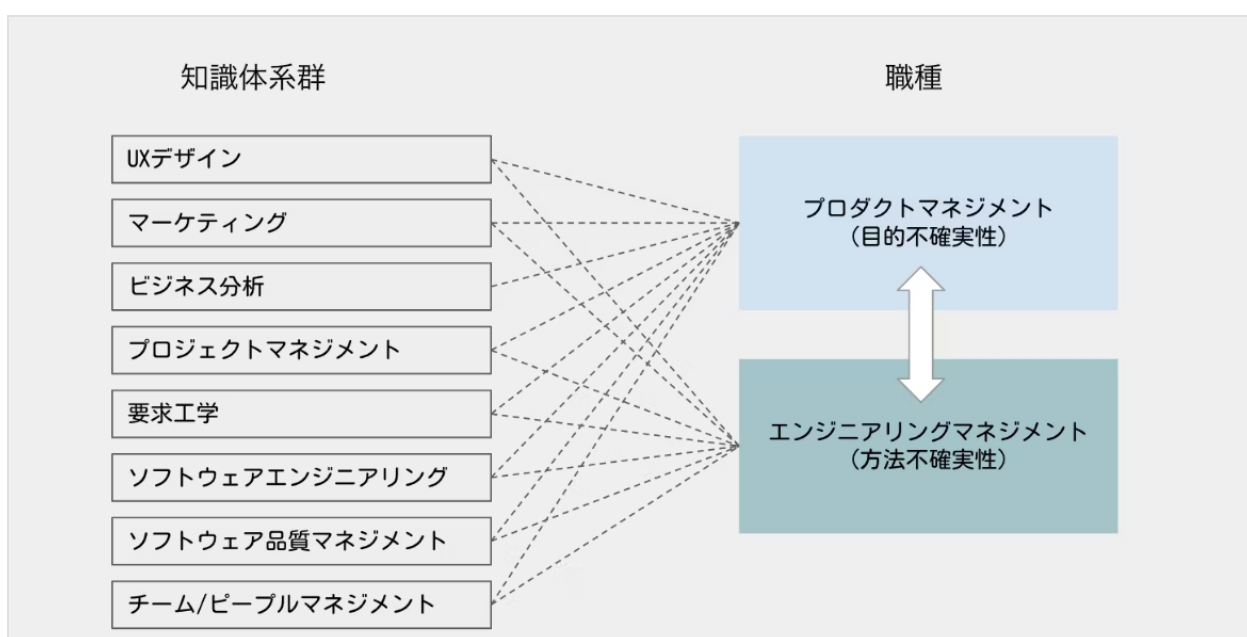
2062

...

す。エンジニアリングマネジメントは、継続的なソフトウェア/デジタル技術関連するシビジネスを開発部隊にとって重要なスキルだからです。

また、もう一角としてプロダクトマネージャと呼ばれる職種を採用する企業も増えてきました。これは、ソフトウェアを構築する際に、「何を作るか」「なぜ作るか」を担当するような職務で、マーケット理解からソフトウェア要件に関する理解までと多彩なスキルが求められます。それに対して、エンジニアリングマネージャは「どうするか」。そこには、「いつまでにつくるか」「どうやって品質と効率を上げるか」「どうやって人を育てるか」といった要素が入ってきます。

なぜ、なにを、どのようにつくるかは連続的なものです。ですから、複数の知識体系から必要に応じて適切な知識や手法を採用することが求められます。



このようなスキルセットは、ソフトウェア開発のための知識体系にも昔から存在します。

たとえば、BABOK/PMBOK/REBOK/SWEBOK/SQuBoKなどの知識体系は、プロダクトマネージャやエンジニアリングマネージャにとって基礎的なスキルやフレームワークを提供してくれるはずで

しかし、SoEに近い領域の自社開発を行うプロダクトマネージャやエンジニアリングマネージャにとって、SoR的な大規模エンタープライズのシステム開発の請負などに比べて、複数の関連会社とのプロジェクトのリソースマネジメントや調達管理をするスキルは内製開発ではほとんど必要とされませんし、複数のステークホルダー分析や、レポート文書や受け入れのための証跡管理のスキルも重視されません。

とても複雑な要件定義書を書いたり 1000や2000のケースを業務分析から漏れ



たストーリーでは必要がないことが多いです。

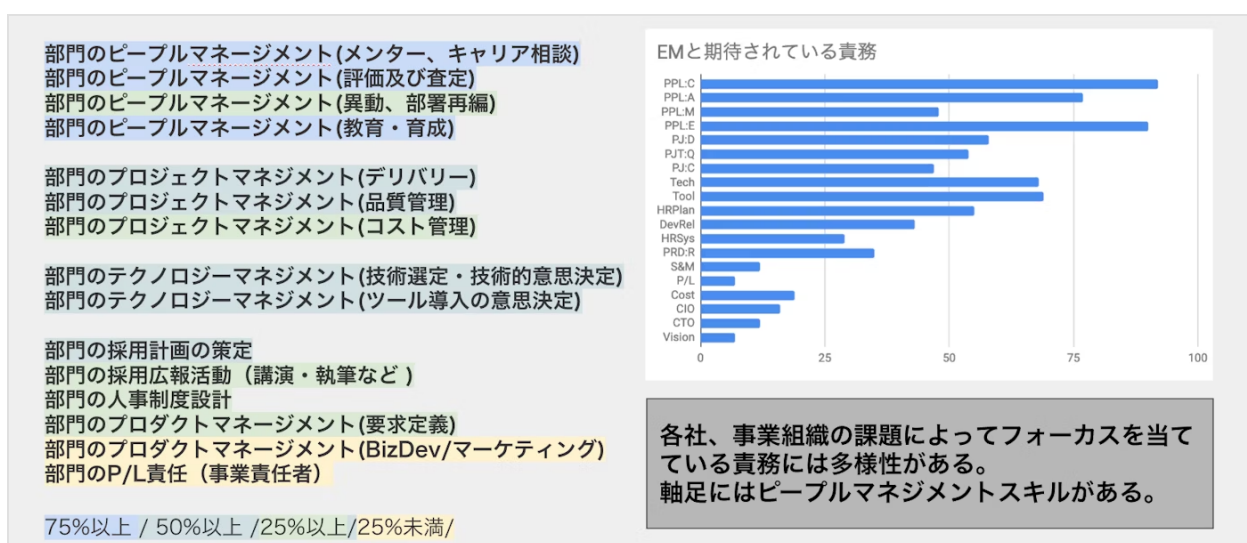
むしろ、いかにして要件自体を小さくするかというマーケティング知識や仮説構築能力がプロダクトマネージャの本懐とも言えるでしょう。

ですが、ソフトウェア要件の書き方の訓練ができていないプロダクトマネージャでは、SaaSやソーシャルゲーム、決済サービスなど高度なトランザクションや複雑な機能要件を定義することができず開発の足を引っ張ってしまいます。

EMもPMも現在は残念ながら、異なる技術や知識体系から自社の今のステージにとって、必要なものを再構成して、適用していく能力が求められることになります。そうであるがゆえに難しく楽しいものではあるのですが。

本稿では、2つの新しいマネジメント職に向けての、この知識体系の柔軟な運用が可能になるような読書のガイドラインを作ることで、健全な業界発展の一助になればと思っています。

EMとはなにか



以前、100名のエンジニアリングマネージャ(および技術系管理職)にアンケートを行い、どのような責務を任せられているのかということ分析しました。

得られた結果としては、ピープルマネジメントのスキルを中心にしながらも、プロジェクトマネジメント、テクノロジーマネジメント、採用活動、要求定義やプロダクトマネジメントといった多岐にわたる責務を担っていることがわかりました。



「弱めのEM定義」と「強めのEM定義」

このような事実から、EMの定義を1つに集約することは難しいことがわかります。

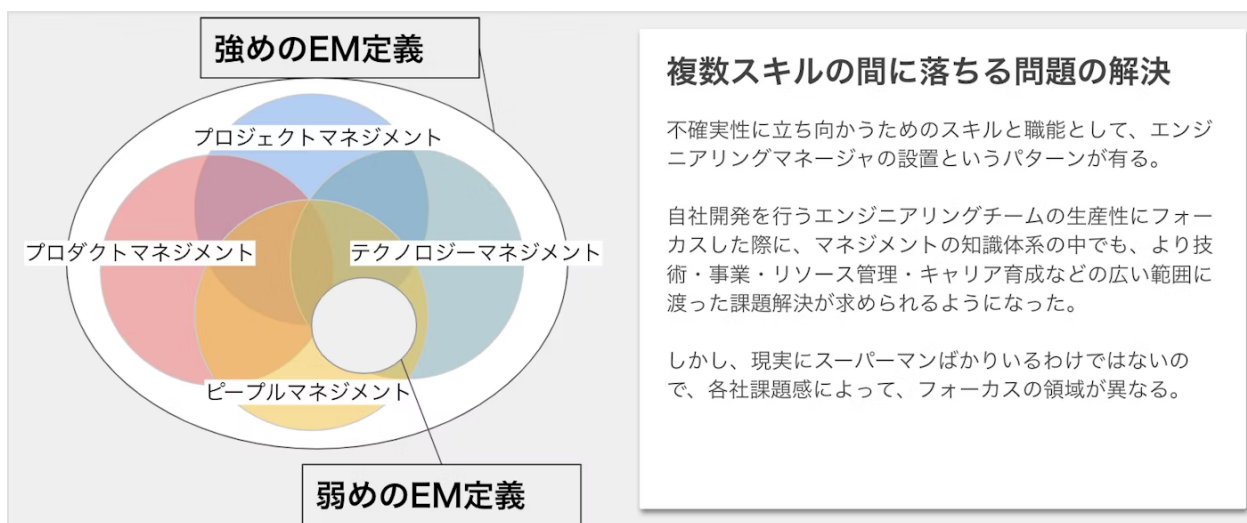
試しに、すべてのEMが求められるような共通部分だけをくくって、「EMの弱い定義」をつくり、関係部分のすべてをさして、より多くのことを求められるものを「EMの強い定義」を作るということを試してみましょう。

「弱めのEM定義」は、「エンジニアのマネジメントスキル」としてのピープルマネジメントスキルとテクノロジーマネジメントのスキルを持っているキャリアになります。

これだけでも、結構難しいものです。というのも、ソフトウェアエンジニアの業界変化ははやく、また転職市場も流動的ですから、常にキャリアになる仕事を提供したり、専門的な知見をもったエルダーとしてアドバイスを行う必要があります。また、その人のキャリアを見越した上での人事評価やスキル評価も難しいため、新しい専門職のチームには「弱めのEM定義」に当てはまるような役職をつけることが少なくありません。

たとえば、フロントエンドエンジニアやアプリエンジニアなど、1つの会社にサーバーサイドエンジニアほど多くいない時期がありました。このような時期に、それぞれのスペシャリティを考慮に入れながら、人事評価や適切なプロジェクト配置を行うことが現実的に難しいという事情から、EMを配置することも少なくありません。

最近では、データサイエンティストや、MLエンジニアなどのキャリアなどを考慮に入れるために専門家集団のエルダーとしてのEMという場合も増えてきました。



一方で、「強めのEM定義」ではどうでしょうか。ピープルマネジメントに加えて、テクノロジーマネジメントのみならず、プロジェクトマネジメントやプロダクトマネジメントの一部の要件定義能力などが求められることもあります。



つまり、SQCD(スコープ、品質、コスト、納期) という4つのトレードオフや課題をマネジメントしながら成果を出すことが求められます。非常に難易度の高い仕事と言えます。

さて、これらすべてに優れた人というのはなかなか採用できるものではありませんし、現実にチームにいないければ分業体制や協力しながら、これらの責務を成し遂げるしかありません。しかし、チームメンバーの期待するマネージャというのは、これらすべてを成し遂げてくれる人であったりします。

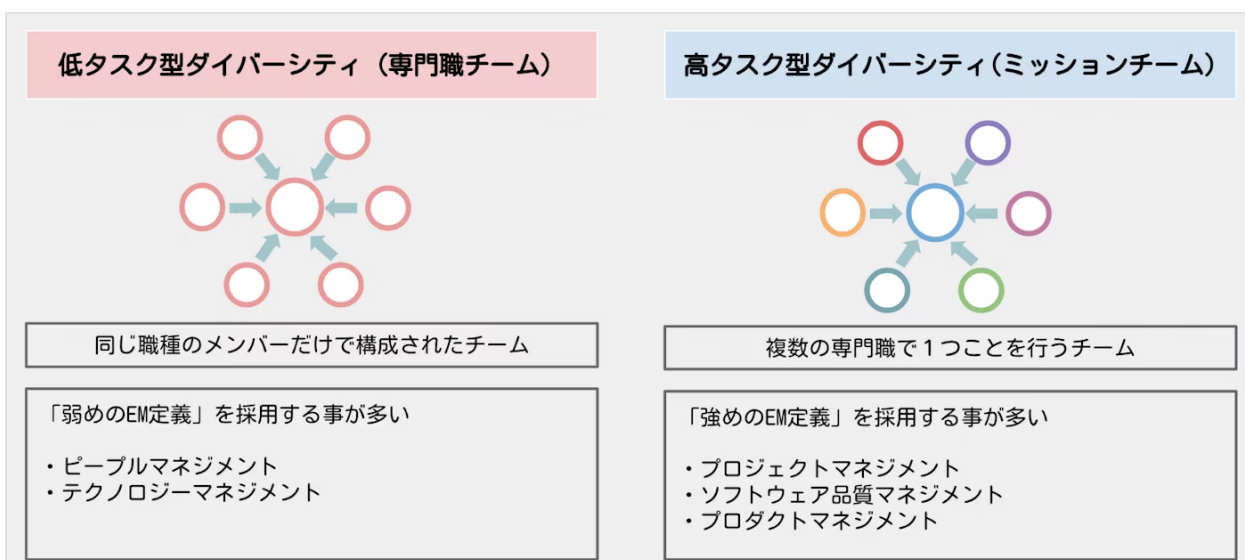
このギャップが「マネージャにスーパーマンを求めてしまう現象」です。

そうはいっても、マネージャ職なのだから「高い給料」をもらっていて、それくらいはできて当然だと考える方もおられるかもしれません。

ソフトウェア開発の現場では、技術的なスキルの高い個人に対しての給与制度も用意されていることが多く、マネージャよりも多くの給与もらっているメンバーを管理することもありふれたことです。(事実、私が初めてマネジメントしたチームは、ほぼ全員が私よりも給与が高い状態でした。)

日本の旧来企業が持っている給与の労使境界がはっきりした人事制度のもとでは、このような考えも成立したかもわかりませんが、ソフトウェアエンジニアの働くチームではこのようなことはなかなか成立しません。

実際に「強いEM」定義と「弱いEM」定義が採用されるケースはどのような違いがあるかというと、チームのタスク型ダイバーシティの高さに由来することが多いです。



タスク型ダイバーシティとは、同一のチームの中にどれだけいろいろなスキルの人が存在するのかを指します。



やすいです。一方で仕事全体のスコープが、市場そのものと接するというより、会社内を顧客とした仕事になりがちです。そのため、短期的な目標に関してのコミットメントに対して意識が弱くなるといったことが起こります。一方、その専門性を発揮して内部人材の育成を強化したい、自社でのキャリアを形成できるようにしていきたいというフェーズに適しています。

タスク型ダイバーシティが高いとマネジメントしているメンバーの専門職のスキルが全部わかるということはありません。でも、大筋わかっていないといけないので、様々なものが求められながらも、価値をとどけるために「強いEM定義」が求められるようになります。プロジェクトやプロダクトのプレッシャーが掛かるため、短期的な目標に対して目が行きがちになります。イノベーションや成果を生みやすい反面、ソフトウェア品質や人材育成から目が離れてしまうことがあります。

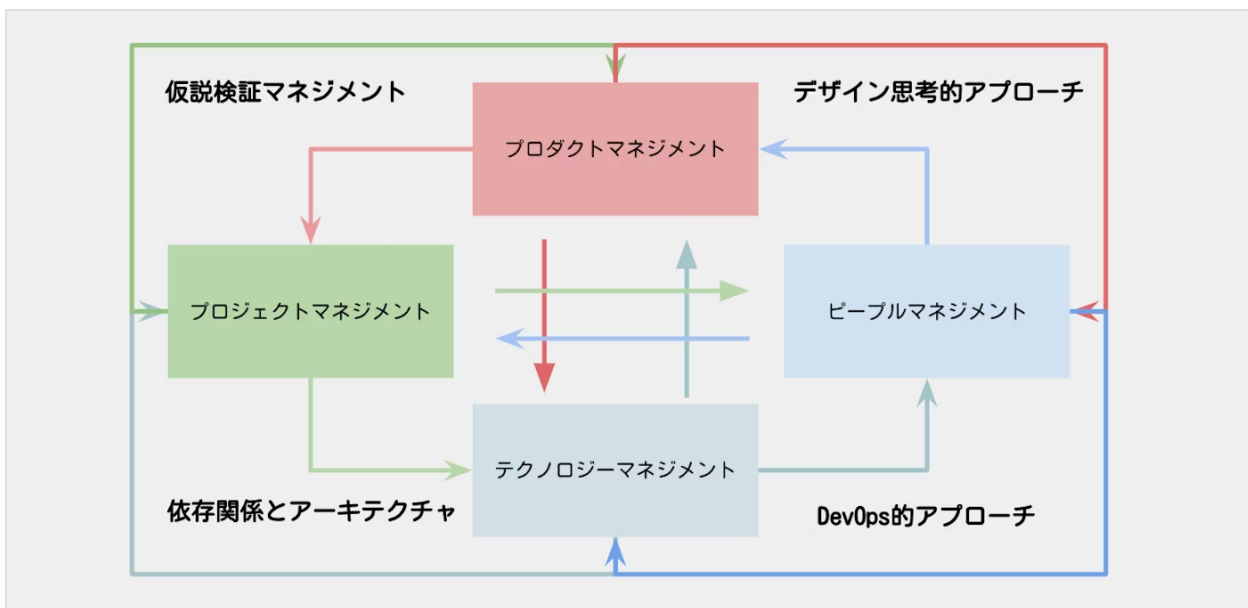
この際のEMは、プロジェクトと品質、文化などのトレードオフを見ながら適切な投資をする能力が求められます。

ですが、そうした場合、「専門職としての経験」を軸にマネジメントを行うことが難しくなるため、結果的にサーバント・リーダーシップや、コーチングなどを通じて、チーム全体が1つのミッションを達成するためのピープルマネジメントスキルが必要になってきます。

各種スキルと読書ガイド

| 大カテゴリ | 少カテゴリ | | |
|--------------|------------------------|--------|--|
| ピープルマネジメント | 1on1/メンタリング | 弱いEM定義 | |
| | ファシリテーション | | |
| | チームング | | |
| テクノロジーマネジメント | ソフトウェア品質 | 強いEM定義 | |
| | XP/DevOps | | |
| | SRE/システムモニタリング | | |
| | アーキテクチャ | | |
| プロジェクトマネジメント | PMBOK | | |
| | 不確実性分析・クリティカルパス | | |
| | CCPM | | |
| | 見積もり | | |
| | アジャイル・リーン | | |
| | CMMI / TSP / PSP | | |
| プロダクトマネジメント | ビジョンマネジメント | | |
| | ソフトウェア要求/仮説検証 | | |
| | プロダクトマーケティング/プロダクトセールス | | |
| | UI/UXデザイン | | |

弱いEMから強いEMにかけての、知識体系をまとめるとこのようになります。自分自身のキャリアや足りない部分などを見つめ直して、インプットするのに使えるとよいかもしれません。



また、それぞれのコアな知識を学んでいくと、学際的な領域にフォーカスが当たった知識が隠れていることが理解できます。

たとえば、ピープルマネジメントとテクノロジーマネジメントの境界線には、DevOps的な思想やツール群が見え隠れします。このトレンドを正しく理解するには、ソフトウェア品質とピープルマネジメントやチームングについて学ぶ必要があるでしょう。

たとえば、プロジェクトにおける依存関係や不確実性について理解すると、交換可能なモジュールやオプションを組み上げて効率的なデザインを行うことのプロジェクトへのポジティブな影響やネガティブな影響を理解できるでしょう。

この他にも、隠れた共通項・学際領域が隠れており、別分野の知識があるのとないのでは理解の深さが異なってきます。

重要なことですが、全てを読む必要はありません

必要に応じて、段階に応じて「このような世界があるんだな」と認識しておき、行き詰まりを感じたときに深堀りをしてみるといいかもしれません。

また各書のリンクは主にshort url作成のためにアフィリエイトリンクを用いています。それらに不快感をお持ちの方は、書名からの検索をお願いします
非アフィリエイトリンクを個別に作成してくださる編集リクエストもお待ちしています。



ピープルマネジメント

1on1/メンタリング

ピープルマネジメントの基本的所作である1 : 1のマネジメントスキルは、究極のところカウンセリングの技法や、精神科医の実践的治験によるところが大きい。実際の所作を獲得することを考えるとこういった書籍が最もわかりやすい。

- コーチングのすべて
- 認知行動療法による対人援助スキルアップ・マニュアル
- マイクロカウンセリング技法
- 傾聴の心理学
- プロカウンセラーの聞く技術
- リフレーミングー心理的枠組の変換をもたらすもの
- 短期間で組織が変わる 行動科学マネジメント

ファシリテーション

1 : 1での行動変容をうながせるようになったら、会議の場でのファシリテーションからプロジェクトファシリテーションなどのステークホルダーの広い場所での行動変容を学んでいくと実際の仕事に活用しやすい。

- ザ・ファシリテーター
- 問題解決ファシリテーター
- ファシリテーターの道具箱
- プロジェクトを変える12の知恵ー ケンブリッジ式 ファシリテーション

チームング

心理的安全性や目標をどのように維持・獲得・達成していくのかの関係性をバラバラに抑えてしまうと、目的と手段が転倒しやすい。なので、OKRなど目標管理と心理的安全性については、同じタイミングで読書すると理解が進む。

- なぜ弱さを見せあえる組織が強いのか
- チームが機能するとはどういうことか
- はじめてのリーダーのための 実践！ フィードバック 耳の痛いことを伝えて部下と職場を立て直す「全技術」



2154

2062

- THE CULTURE CODE
- 具体と抽象

テクノロジーマネジメント

ソフトウェア品質

ソフトウェアプロジェクトの成否を決めるほど強い要素でありながら、しばしば理解が及んでいないことがあるのがソフトウェア品質に関する知識です。様々なファクトとデータ分析が積み重なっている分野でもあり、これらの知見はエンジニアリングマネジメントにとって必要不可欠と言えます。

- ソフトウェア品質知識体系ガイド -SQuBOK Guide-(第2版)
- データ指向のソフトウェア品質マネジメントーメトリクス分析による「事実にもとづく管理」の実践
- 初めて学ぶソフトウェアメトリクス~プロジェクト見積もりのためのデータの導き方
- ソフトウェア品質の経済的側面

XP/DevOps

XPの習慣とDevOps的なアプローチは、ソフトウェア品質のための、協調・創造の文化と自動化の技術、プログラミング的習慣によって構成されている。直接的なファクトが得られづらいが、実践を積み重ねることによって高度な技術チームを作り上げることができる。

- テスト駆動開発
- モブプログラミング・ベストプラクティス ソフトウェアの品質と生産性をチームで高める
- The DevOps ハンドブック 理論・原則・実践のすべて
- LeanとDevOpsの科学[Accelerate] テクノロジーの戦略的活用が組織変革を加速する
- レガシーコードからの脱却ーソフトウェアの寿命を延ばし価値を高める9つのプラクティス

SRE/モニタリング



2154

2062

ソフトウェア品質は、事前と事後の品質が存在する。リリース事後において発見されるバグというのは、ソフトウェアライフサイクルの中で非常に大きい割合を占める。以下にして、安定した運用を実現するのかという技術や文化、習慣の理解がないと、事前品質と事後品質のトレードオフやメトリクスが理解できなくなってしまう。

- SRE サイトリライアビリティエンジニアリング —Googleの信頼性を支えるエンジニアリングチーム
- 入門 監視 —モダンなモニタリングのためのデザインパターン
- Infrastructure as Code —クラウドにおけるサーバ管理の原則とプラクティス
- 詳解 システム・パフォーマンス

アーキテクチャ

継続的にソフトウェアの開発効率を維持し、バグの起こりにくい・起きてもクリティカルになりにくいといった性質を実現するためには、アーキテクチャの素養がなければならない。ソフトウェアプロジェクトマネジメントの場合、アーキテクチャやソフトウェア品質に関する理解があるかないとでは、打ち手の幅が全然変わってくる。モジュール化の理論は、プロジェクトマネジメントと同じく不確実性とオプションの数理で構成されていることがわかってくると、プロジェクトマネジメントとソフトウェアアーキテクチャの隣接点が浮かび上がってくる。

- ソフトウェアシステムアーキテクチャ構築の原理 第2版
- Clean Architecture 達人に学ぶソフトウェアの構造と設計
- エンタープライズアプリケーションアーキテクチャパターン
- アプリケーションアーキテクチャ設計パターン
- .NETのエンタープライズアプリケーションアーキテクチャ第2版 .NETを例にしたアプリケーション設計原則
- 進化的アーキテクチャ —絶え間ない変化を支える
- Design It! —プログラマーのためのアーキテクティング入門
- 実践ドメイン駆動設計 (Object Oriented SELECTION)
- モジュール化—新しい産業アーキテクチャの本質 (経済産業研究所・経済政策レビュー)

プロジェクトマネジメント

PMBOK



2154

2062

プロジェクトファシリテーションの知見やチーミングが理解できていると、プロジェクトマネジメントのプラクティスの理解が早くなる。PMBOKなどの知識体系はとても広い。ソフトウェアだけではないプロジェクトマネジメントの一般的な性質について理解してから、ソフトウェアに戻ってきたほうがポイントが抑えやすい。

- 図解入門よくわかる 最新PMBOK第6版の基本
- 図解 これ以上やさしく書けない プロジェクトマネジメントのトリセツ
- プロジェクトマネジメントの基本 この1冊ですべてわかる

不確実性分析・クリティカルパス

プロジェクト管理の手法の背後には、統計的な不確実性の取り扱い方や、依存関係の取り扱い方と言った統計数理と離散数理がついてまわる。自分たちがリスクという不確実性の管理を行っているという見方を広げていくと、プロジェクトマネジメントのレベルが一弾上がるはず。

- 不確実性分析
- 計画の科学
- ピープルウェア
- 熊とワルツを リスクを愉しむプロジェクト管理

CCPM

CCPMは、納期に向けてのプロジェクト管理でありながら、バリューストリームの最適化のToCなどの理論を背景に持つ管理手法。バッファを統合し、バッファの消費率でプロジェクトの健全性をチェックするなど、不確実性を見える化して進むプロジェクト管理の重要手法。

- クリティカルチェーン—なぜ、プロジェクトは予定どおりに進まないのか?
- 進む! 助け合える! WA(和)のプロジェクトマネジメント——プロマネとメンバーのためのCCPM理論
- アジャイルCCPM: プロジェクトのマネジメントを少し変えて組織全体のパフォーマンスを大きくのばす
- 品質と生産性を重視したソフトウェア開発プロジェクト技法—見積り・設計・テストの効果的な構造化

見積み



2154

2062

ソフトウェアプロジェクトにおける不確実性の原因の1つは、見積もり誤差から生じる。この見積もりにというものについて、人々はどのような歴史を持ち、どのような手法を分析してきたかを理解するとソフトウェアプロジェクト管理の半分が手のうちに入る。

- アジャイルな見積りと計画づくり ～価値あるソフトウェアを育てる概念と技法
- ソフトウェア見積り
- ソフトウェア見積りのすべて 第2版 ー現実に即した規模・品質・工数・工期の予測

アジャイル/リーン

これまでのソフトウェアプロジェクト管理の流れから、どのようにしてスクラムやアジャイルなどの手法が生まれてきたかを理解すると、ポイントがおさえやすい。逆に最初からスクラムだけを学ぼうとしたり、リーンだけを学ぼうとすると躓くことがある。従来のプロジェクトマネジメントの知見を理解してそのうえにアジャイル開発を捉えたほうがいい。

- アジャイルサムライ
- 大規模スクラム Large-Scale Scrum(LeSS)
- エssenシャル スクラム: アジャイル開発に関わるすべての人のための完全攻略ガイド (Object Oriented Selection)
- アジャイル開発とスクラム~顧客・技術・経営をつなぐ協調的ソフトウェア開発マネジメント
- リーン開発の本質
- アジャイルイントロダクション：Agile開発の光と影
- 知的創造企業

CMMI

経験主義的チームの管理手法は、スクラムだけではない。ケイパビリティではなく成熟度モデルと経験主義的工数管理がCMMI/PSP/TSPなどの知識体系として存在したことを思い出していくと、よりソフトウェアプロジェクトマネジメントの理解が進む。

- 図解 はじめての「開発のためのCMMI」とプロセス改善 第2版
- PSPガイドブック ソフトウェアエンジニア自己改善



ビジョンマネジメント/ビジネス分析

プロダクトマネジメントは、プロダクトの各フェーズごとに不確実性のジャンルが異なります。

顧客理解であったり、コンセプトであったり、ビジネスモデルであったり、デザインやマーケティングであったり。

多くの場合、フェーズと自分の得意分野がミスマッチしてしまっても、新しいステージでの学習が進まず成長が飽和してしまうことがあります。自分の事業のフェーズ認識とそのビジョン共有がプロダクトマネジメントの最も難しいポイントです。

- リーン・スタートアップ
- 起業の科学 スタートアップサイエンス
- ビジネスモデルの教科書: 経営戦略を見る目と考える力を養う
- ビジネスモデル for Teams 組織のためのビジネスモデル設計書
- バリュー・プロポジション・デザイン 顧客が欲しがる製品やサービスを創る
- INSPIRED 熱狂させる製品を生み出すプロダクトマネジメント

ソフトウェア要求/仮説検証

また、現在では単純だがWebメディアだけでなく、複雑な業務用件を満たした上で複数社に同じサービスを提供するようなB2B SaaS型/B2B2C型マッチングのサービスを開発していく企業が増えました。

これまで、画面仕様書をざっくり書いて伝わっていたことが、正確に要求を分析した上で定義していくことが求められるようになりました。また、まだ見ぬ複数の顧客に提供しても勝ちがあるかどうか重要なファクターですから、仮説検証的に「最小限度の価値のある単位」に分割して、顧客の応答を求めていくような開発要件定義も必要になります。

このあたりは「開発キャリア型のプロダクトマネージャ」か「BizDevキャリア型のプロダクトマネージャ」で、スキルにギャップが現れるところです。

- 【改訂第2版】【入門+実践】要求を仕様化する技術・表現する技術 ~仕様が書けていますか？
- ソフトウェア要求と仕様（電子版）：実践，原理，偏見の辞典
- 要求工学実践ガイド: REBOKシリーズ2
- アジャイル開発の本質とスケールアップ 変化に強い大規模開発を成功させる14のベストプラクティス



- 正しいものを正しくつくる プロダクトをつくるとはどういうことなのか、あるいはアジャイルのその先について
- モデルベース要件定義テクニック

プロダクトマーケティング/プロダクトセールス

プロダクトマネジメントのフェーズが、PMFに到達する前後でセールスラインやプロダクトマーケティングがプロダクトマネージャの主な関心事になります。自社のセールスモデルかマーケティングモデルが、フィットしているかを検証しながら、グロースをすすめるための基礎的な知識をもつ必要があります。

これまで、0-1で開発チームとの時間を長く過ごしたプロダクトマネージャがマーケティングチームとつきっきりになるため、組織問題を生み出しやすい時期でもあります。

- THE MODEL (MarkeZine BOOKS)
- カスタマーサクセス——サブスクリプション時代に求められる「顧客の成功」10の原則
- サブスクリプション——「顧客の成功」が収益を生む新時代のビジネスモデル
- [改訂4版] グロービスMBAマーケティング
- マーケティングリサーチの論理と技法
- 最新マーケティングの教科書2019
- プロダクトマネジャーの教科書

UXデザイン/ユーザビリティ

プロダクトマネージャとエンジニアリングマネージャにとって、デザインやUXの知識はプロダクトの成否を分けるほど重要な知識です。

デザインの思考法とUIコンポーネントやユーザビリティ、総合的なUX

- ユーザーエクスペリエンスの測定: UXメトリクスの理論と実践
- ユーザビリティエンジニアリング: ユーザーエクスペリエンスのための調査、設計、評価手法
- デジタルプロダクトのためのデザインシステム実践ガイド
- エンジニアのためのデザイン思考入門
- ユーザーインタビューをはじめよう
- デザイン組織のつくりかた
- デザインリーダーシップ



2154

2062

- ノンデザイナーズ・デザインブック [フルカラー新装増補版]
- デザインング・インターフェース 第2版 —パターンによる実践的インタラクションデザイン
- マイクロインタラクション —UI/UXデザインの神が宿る細部
- Running Lean
- 一人から始めるユーザーエクスペリエンス

おわりに

繰り返しになりますが、すべての本を読む必要はありません。
マネジメントは様々なスペシャリストと対話をしながら適切な意思決定をしていくための機能です。
なので、スペシャリストとの会話ができるような各分野への関心とリスペクトを持ち続けることが最も重要です。





 1



関連記事 Recommended by

エンジニアリングマネージャーになる前に知りたかった考え方

by kyntk

エンジニアリングマネージャのスキル習得

by newta

エンジニアリング組織の仕事を考えることで、EMが学ぶべきことを体系化してみた

by expajp



2154

2062

上司（EM）にモヤッとしたときに読む記事＝モヤッとさせる上司（EM）になりたくない人のためのヒント

by m3hiro3

サブスク事業に特化した決済代行×会員サイト構築する方法

PR 株式会社ROBOT PAYMENT

公開型生体認証基盤を活用して購買体験のアップデートを仕掛ける、日立的PM

PR 株式会社 日立製作所

コメント

この記事にコメントはありません。

T コメントする



プレビュー

コミュニティガイドラインに基づき、良識ある内容を心がけましょう。

テキストを入力

0B / 100MB

投稿する

How developers code is here.



| ガイドとヘルプ | コンテンツ | SNS |
|--------------|---------------|---------------|
| About | リリースノート | Qiita（キータ）公式 |
| 利用規約 | 公式イベント | Qiita マイルストーン |
| プライバシーポリシー | 公式コラム | Qiita 人気の投稿 |
| ガイドライン | アドベントカレンダー | Qiita（キータ）公式 |
| デザインガイドライン | Qiita 表彰プログラム | |
| ご意見 | API | |
| ヘルプ | | |
| 広告掲載 | | |
| | | |
| Qiita 関連サービス | 運営 | |
| Qiita Team | 運営会社 | |
| Qiita Jobs | 採用情報 | |
| Qiita Zine | Qiita Blog | |
| Qiita 公式ショップ | | |



2154

2062