
機能追加とリーダー業務 との類似性

大吉祥寺.pm 2025 09/06
株式会社400F Rinchoku

アジェンダ

- 自己紹介
 - 最初に
 - リーダーとエンジニアの違い
 - 機能追加の一般的な流れ
 - 何が類似性があるのか
 - リーダーとエンジニアの決定的な違い
 - 最後に
-

自己紹介

名前: Rinchoku

所属: 株式会社400F

X: @stupid_owl

趣味: 宝石鑑賞、ハンドメイド、ジョギング

その他

- スタッフ活動

- オブザーバビリティカンファレンスTOKYO 10/27
- SRE Kaigi 2026コアスタッフ 1/31
- SRE NEXT 2026 Coming soon

- 登壇

- PHP Conference 2025
- PHPer Kaigi 2025



自己紹介

好きな俳人:「秋元不死男」

代表的な句

- 鳥わたるこきこきこきと罐切れば
- へろへろとワンタンすするクリスマス

※今日の朝wikipediaで知った



最初に

本トークの立ち位置

- 「メンバー」から「リーダー」になった方へのメッセージ
 - 「メンバー」時よりパフォーマンスを発揮できない人
 - リーダーになって業務で不安がある人

※以降「メンバー」を「エンジニア」と呼称して進めます



免責

- 本トークは私が経験したことベースの発表です
 - 科学的な根拠や論文などの明確なデータはありません
 - 中小企業での「リーダー」を想定
 - エンジニア総数も少ないため、早めにリーダーをできる人材を作りたい
 - マネジメント業務(一部)も開発業務どちらも行う
 - マネジメント: スケジュール管理、ピープルマネジメント、工数管理、etc...
 - 開発業務: 要件定義、実装、テスト、技術調査、etc...
 - 1チーム4~5人程度
-

最初に

みなさんが「リーダー」という役職になったきっかけって何ですか？



最初に

みなさんが「リーダー」という役職になったきっかけって何ですか？

- 上長から勧められたから
- 給料が増えるから
- 将来マネジメント職に興味があり、そのキャリアパスの中間地点
- 「スペシャリスト目指すならリーダー経験も必要だよ」と言われたから
- 人がいないから
- 何となく
- etc...



最初に

リーダーをやって感じること

最初に

リーダーをやって感じること

- コーディングより、人やプロダクトと向き合う時間が増えた
- 一日のMTG量が激増した
 - 1 on 1、リーダー以上のMTG、事業部とのMTG
- コーディングする時間を作りたい！！
- コーディング力・技術力が下がっていないか 心配になる
- 自分のやり方があっているかわからない
- etc...



最初に

リーダーになって半年後の私が感じたこと

最初に

リーダーになって半年後の私が感じたこと

- いろいろやることが増えた気がした
 - その一方で、どうしていけばいいかわからない。。。
 - なんか今までのやり方ではうまくいかないな。。。
 - コード書く時間あまりないな。。。
-

最初に

リーダーになって1年後の私が感じたこと

最初に

リーダーになって1年後の私が感じたこと

- 今までやった開発の進め方を応用したら、いい感じじゃん
 - よく見ると、リーダー業務も今までの延長戦かも
 - コード書きたいな。。。
-

最初に

リーダーになって1年後の私が感じたこと

- 今までやった開発の進め方を応用したら、いい感じじゃん
 - よく見ると、リーダー業務も今までの延長戦かも
 - コード書きたいな。。。
-

「リーダー」と「エンジニア」では違いはあるよね？

リーダーとエンジニアの責務の違い

	エンジニア	リーダー
求められること	プロジェクト・タスクの自走	チームが目標に同じ方向を向いて進んでいること
責任範囲	個人・プロジェクト	チーム・プロダクト
影響範囲	チーム・開発	チーム・プロダクト関係者
評価指標	品質・納期・コスト(QCD)	KPI・ROIの一部
行動による成果	短期(数か月後)	中期~長期(半年~数年)

リーダーとエンジニアの求められるスキルの違い

エンジニア

- 技術的な課題解決力
- 技術的な情報収集力
- アーキテクチャ理解力
- コーディング力
- 専門性の知識
- etc...

リーダー

- プロジェクト管理
 - スケジュール管理
 - 組織牽引力
 - リスクマネジメント
 - コスト管理
 - プロダクト理解力
 - ピープルマネジメント
 - etc...
-

リーダーとエンジニアの違い

- スキルや責務は違うところが多く感じる
 - 会社内でもキャリアパスが違う
 - マネジメントとスペシャリストの2方向に分離
 - 上記の認知によって、未経験者には障壁やギャップの違いを感じる
-

新米リーダーの悩み

- 今までと傾向が違いすぎて、何から手を出すべきかわからない
- 「何を」やるかは分かった。「どう」すればいいかわからない
- とりあえず今まで通りやってみたけど、うまくいかない
- etc

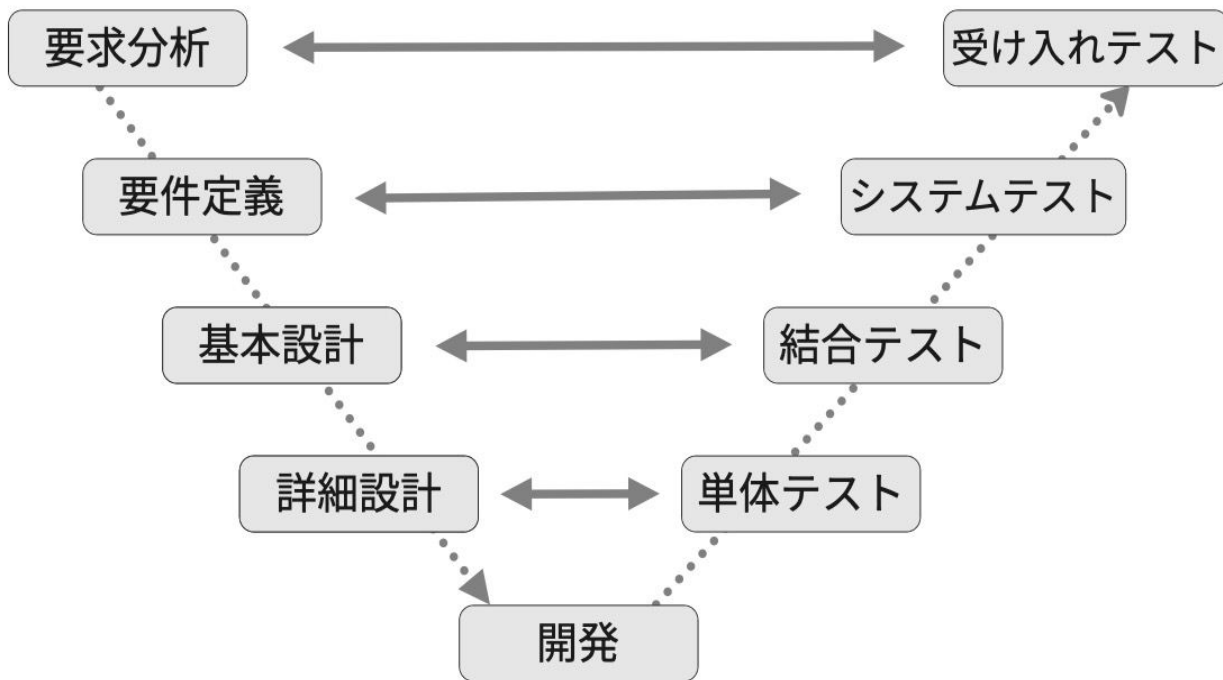


あなた/わたしはなぜリーダーになれたのか

上長がリーダーになれる能力がある と思ったから

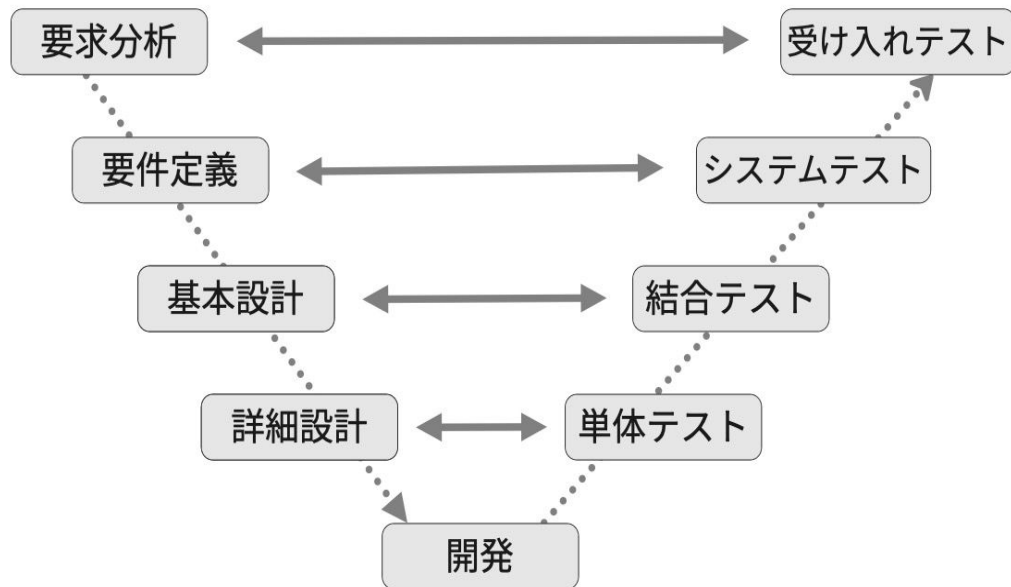
今までの業務経験でリーダーが行えると**信頼を得た** から

機能追加の一般的な流れ



機能追加の一般的な流れ

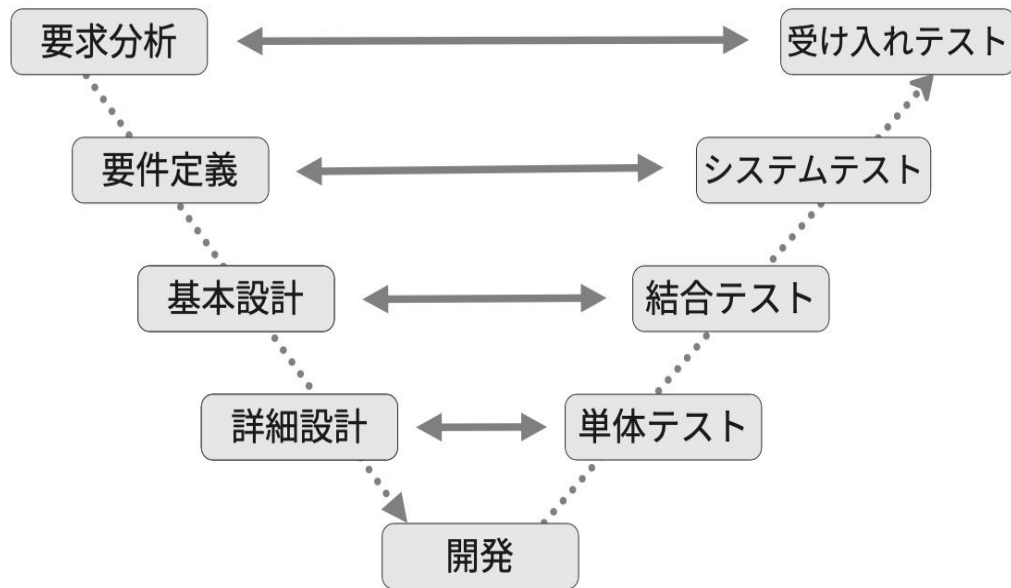
- エンジニアは各ステップで成果物を作成
- 下流を経験後、少しずつ上流を経験していくことが一般的



機能追加の一般的な流れ

ここでの疑問

- どうこれがリーダー業務と関連づくの？
- リーダー業務の一部は内包してるけど、これだけでリーダーはできないでしょ？

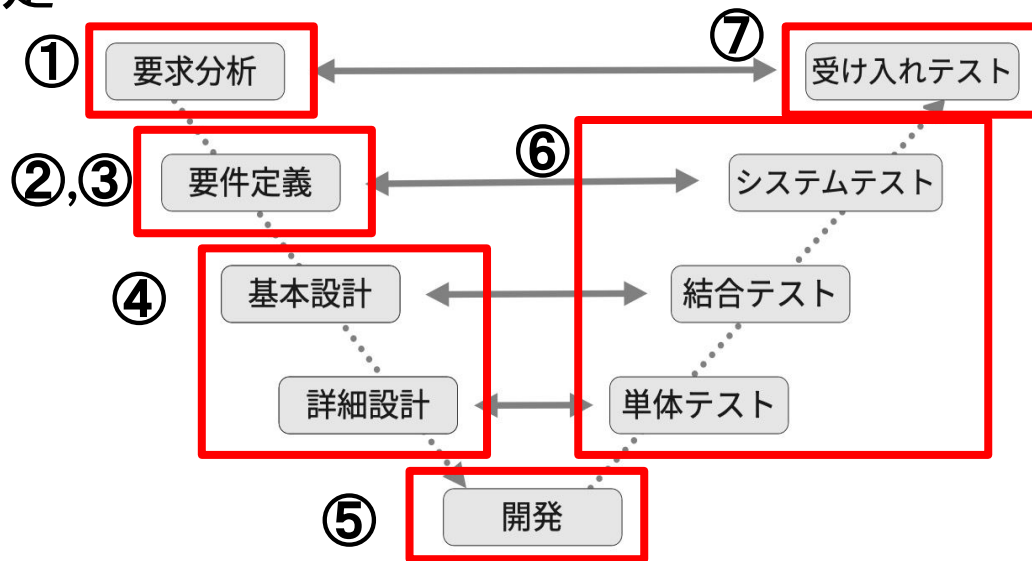


機能追加の進め方一例

1. ユーザーの課題の把握
 2. 課題の解決方法の選定
 3. 影響範囲の調査
 4. 実行手順の整理
 5. 実行
 6. 動作確認
 7. リリース
-

機能追加の進め方一例

1. ユーザーの課題の把握
2. 課題の解決方法の選定
3. 影響範囲の調査
4. 実行手順の整理
5. 実行
6. 動作確認
7. リリース



機能追加の進め方一例

このフローを少し抽象化して考えると。。。

- | | | |
|---------------|---|------|
| 1. ユーザーの課題の把握 | } | 現状把握 |
| 2. 課題の解決方法の選定 | | |
| 3. 影響範囲の調査 | } | 計画 |
| 4. 実行手順の整理 | | |
| 5. 実行 | } | 実行 |
| 6. 動作確認 | | |
| 7. リリース | } | 確認 |
-

機能追加の進め方一例

- 1. ユーザーの課題の把握
 - 2. ユーザーの課題
- } 現状把握

- 3. の
- 4. 影
- 5. 実

PDCAのサイクル
と同一視できる

- 6. 実行
- 7. 動作確認
- 8. リリース

} 実行
} 確認

機能追加もリーダー業務も似ている

機能追加は大雑把に考えると

- プロダクトの既存機能に手を加えて、ユーザーによりよい価値を提供する
-

機能追加もリーダー業務も似ている

機能追加は大雑把に考えると

- プロダクトの既存機能に手を加えて、ユーザーによりよい価値を提供する

リーダー業務を大雑把に考えると

- チームや周辺環境に手を加えて、プロジェクトを成功に導く
-

機能追加もリーダー業務も似ている

機能追加は大雑把に考えると

- プロダクトの既存機能に手を加えて、ユーザーによりよい価値を提供
対象が変わるだけで、
リーダー やりたいことは変わっていない
- チームや周辺環境に手を加えて、プロジェクトを成功に導く



機能追加もリーダー業務も似ている

根本を突き詰めると、PDCAと似ている

機能追加もリーダー業務も似ている

根本を突き詰めると、PDCAと似ている

リーダーの最大の目標: プロジェクトをチームで成功へ導く

機能追加もリーダー業務も似ている

根本を突き詰めると、PDCAと似ている

リーダーの最大の目標: プロジェクトをチームで成功へ導く

そのためには。。。。

- チームが同じ目標に向かって進めるようにする
 - プロジェクト進行上の課題を取り除く
 - メンバーの調子が把握しておく
 - etc...
-

機能追加もリーダー業務も似ている

やることの大枠は同じ。

別軸と捉えるから失敗しやすい

機能追加もリーダー業務も似ている

やることの大枠は同じ。

別軸と捉えるから失敗しやすい

→「抽象化」がとても大事

機能追加もリーダー業務も似ている

やることの大枠は同じ。

別軸と捉えるから失敗しやすい

→「抽象化」がとても大事

→自分の得意領域(成功パターン)に持っていく

→無数に経験した機能追加に成功パターンは詰まっている

何が類似性があるのか

プロジェクトのスケジュール管理

- 機能追加で行ったこと
 - 見積もりと実績の乖離を最小限にする
 - タスクを細かく分ける
 - 事前に外部影響や仕様上時間がかかりそうなところをリストアップ
 - リーダーとしてのスケジュール管理も考えは同じ
 - 一つ一つのタスクの見積もりと実績の乖離が起きた時に対策を講じる
 - 事前に進行上に障害をリストアップし、対応を考える
-

何が類似性があるのか

ピープルマネジメントの場合

- 機能追加の場合
 - 自分の体調管理や、開発しやすい状態を維持
 - 「知っていること」と「知らないこと」の把握
 - リーダーのピープルマネジメントに置き換えると
 - 各メンバーの開発しやすい状況(モチベーションや体調など)の理解
 - チームとして開発しやすい状況への改善
 - 場合によってはリーダーだからこそ環境を整えるという一歩もあり
-

何が類似性があるのか

そうは言っても、対象が変わるとやっぱり難しいよ。。。

何が類似性があるのか

そうは言っても、対象が変わるとやっぱり難しいよ。。。

機能追加でも似たことはあったはず

何が類似性があるのか

そうは言っても、対象が変わるとやっぱり難しいよ。。

機能追加でも似たことはあったはず

- 自分の知識内で施策をいくつかリストアップして選択
 - より詳しくそうな人に相談する
 - 本やXとかで情報を集める
 - 自分がやってみたいと思ったことを取り入れる
-

リーダーとエンジニアの決定的な違い

ここまで、「類似性」について話してきました

ただ一方で、決定的な違いも一つある



リーダーとエンジニアの決定的な違い

「責任」と「権限」の所在

- 責任

- 説明責任: 関係者への最終成果物および過程で発生した事象を伝える責任
- 実行責任: タスクの達成に対する責任

- 権限

- 実行権限: タスク進行の実行範囲で与えられる限定された権限
 - 実行権限を持つ＝対象のタスクを進める



リーダーとエンジニアの決定的な違い



マネージャー

説明責任
実行責任
実行権限



リーダー



エンジニア

リーダーとエンジニアの決定的な違い


マネージャー

説明責任
実行責任
実行権限


リーダー

説明責任
実行責任
実行権限


エンジニア

実行権限を委譲
実行するうえでマネージャーへの説明の義務もあるため、説明責任も追加される
※場合によって、実行責任も委譲することもある

リーダーとエンジニアの決定的な違い



説明責任
実行責任

マネージャー



説明責任
実行権限

リーダー



エンジニア

リーダーとエンジニアの決定的な違い



マネージャー

説明責任
実行責任



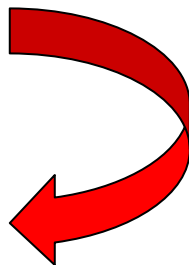
リーダー

説明責任
実行責任
実行権限



エンジニア

説明責任
実行権限



実行権限の委譲

リーダー: エンジニアが完了した成果物に対する責任を持つため、実行責任を持つ
エンジニア: リーダーへのタスクの状況を説明するため、説明責任が追加

リーダーとエンジニアの決定的な違い



マネージャー

説明責任
実行責任



リーダー

説明責任
実行責任



エンジニア

説明責任
実行権限

リーダーとエンジニアの決定的な違い



マネージャー

説明責任: **他部署や株主、顧客**などに対する説明
実行責任: **組織・市場**を見通しての成果物の責任



リーダー

説明責任: **プロジェクト関係者**に対する説明
実行責任: **プロジェクト内の成果物**の責任



エンジニア

説明責任: **開発関係者**に対する説明
実行権限

リーダーとエンジニアの決定的な違い



説明責任: 他部署や株主、顧客などに対する説明
実行責任: 組織・市場を見通しての成果物の責任

各役職の「説明責任」の認識を間違えると
間違った「説明」となる

大行具は、ノロノエノド内ノ成果物の具は

リーダー



エンジニア

説明責任: 開発関係者に対する説明
実行権限

責任の意味・範囲の違いによる「説明」の違い

- リーダーは「マネージャー」と「エンジニア」の間にいることで、**どちらにも沿った状況把握**が必要になる
 - 特にエンジニアに確認する場合は、位置による認識合わせも大事
 - 認識がすれ違うと「マイクロマネジメント」が行われていると感じさせてしまう
 - この説明責任を達成できるように、リーダー業務を意識することが重要
 - 実行責任の達成のために、いい感じに「委譲する」のが大事
 - 直接のタスクは「委譲」、チームの改善は「自身」と棲み分けもあり
-

最後に

- 未知と思ったものも**自分の成功パターンに持っていく**
 - 成功パターン自体は、過去の経験が教えてくれる
 - 「未知」と思わず、似ているところを見つける
 - そのためにも、「抽象化」は重要
 - リーダーの**「説明責任」と「実行責任」**を認識する
 - 流れのまま行くと、「リーダー」ではなく「リーダーの役職を持ったエンジニア」になってしまう
 - その範囲内で自分なりの「目的」「目標」を設定することで、より進めやすくなる
-

ここで一句

リーダーへ そぞろ跳ね除く 我が軌跡

