

最近読んだ技術書を簡単紹介



Press Space for next page →



自己紹介

- 📄 飯野陽平 ([wheatandcat](#))
- 🏢 会社: [合同会社UNICORN](#) 代表社員
- 📖 Blog: <https://www.wheatandcat.me/>
- 🛠️ 今までに作ったもの
 - [memoir](#)
 - [OOMAKA](#)
 - [MarkyLinky](#)

今回のスライド

- 技術系の知識のインプット元の情報をアウトプットしていないと思い、まとめました
- 自分は主に以下の3つから情報を得ている
 - 技術書
 - Podcast
 - Webサイト
- 2022～2024年で読んで面白かった技術本を簡単に紹介

オブジェクト指向UIデザイン

ー使いやすいソフトウェアの原理

- オブジェクト指向 UI（OOUI）の解説本
- 論理的にUI設計を行うための手法を学べるので、エンジニアがUI設計を理解するのに最適
- 中盤からは特定の要件からオブジェクト指向 UIを設計するクイズ形式で進む感じになっているので、実践的にUIを学ぶことができる

オブジェクト指向 UIデザイン

使いやすいソフトウェアの原理

ソシオメディア株式会社、上野 学、藤井幸多

〔著〕

上野 学

〔監修〕

Object-Oriented User Interface

銀の弾丸、OOUI。

タスクからオブジェクトへの転回による
操作性と開発効率の劇的な向上

手を動かして身につける18の実践演習付き

技術評論社

SCRUM BOOT CAMP THE BOOK

スクラムチームではじめるアジャイル開発

- スクラムの基本的なイベントを学べる
- マンガ形式 + 解説で進行されるのでスクラムの解説本の中では、かなり読みやすい
- スクラムについて深く学ぶ本ではないが、全体像を把握するのに良い

SHOENISHA

スクラムチームで
はじめる
アジャイル開発

増補
改訂版

SCRUM BOOT CAMP THE BOOK

スクラム・ブート・キャンプ ザ・ブック

〔著〕

Naoito Nishimura 西村 直人

Miho Nagase 永瀬 美穂

Ryutaro Yoshida 吉羽 龍太郎



はじめて「スクラム」を
やることになったら
読む本！

理論だけでは
終わらない“実践”の手引き
架空の開発現場を題材に
実際のプラクティスを詳説！

The DevOps ハンドブック

理論・原則・実践のすべて

- アジャイルからの文脈でDevOpsについて知識をつけたかったので読んでみた
- 前のページで紹介したスクラムの本は製品リリースまでの話を中心に運用開始後の話は無い
- DevOpsは運用開始後に、どうサイクルを回すのかの話が中心

THE DevOps HANDBOOK

ジーン・キム、ジェズ・ハンブル、パトリック・ドボア、ジョン・ウィリス
榊原彰 監修 長尾高弘 訳 定価:[本体3200円+税]

The DevOps ハンドブック

理論・原則・実践のすべて

日経BP社

DevOps改革を
《迅速に・確実に・安全に》
実践するための必読書

「TheDevOps 逆転だ!」の「3つの道」の全貌が明らかに

チームトポロジー

価値あるソフトウェアをすばやく届ける適応型組織設計

- DevOpsの発展型の文脈で読んでみた
- 開発のサイクルを適切に回すための最適になチームの形について解説している
- スクラム、DevOpsの話の先にある、組織の成長、システムの大規模化の概念についての対策本
- 本の中で各役割のチームや各コミュニケーションの定義に名前をつけていて、本を読んだもの同士で共通認識が持ちやすい

価値あるソフトウェアを
すばやく届ける
適応型組織設計

訳 著
吉羽 龍太郎 永瀬 美穂 原田 騎郎
マシユー・スケルトン
マニユエル・パイス

Organizing Business and Technology Teams for Fast Flow Team トポロジー Topologies

米国で絶賛！
IT組織論の
流れを変える
話題作！

真のパフォーマンスを
引き出すカギは
チームタイプと
インタラクションモード！

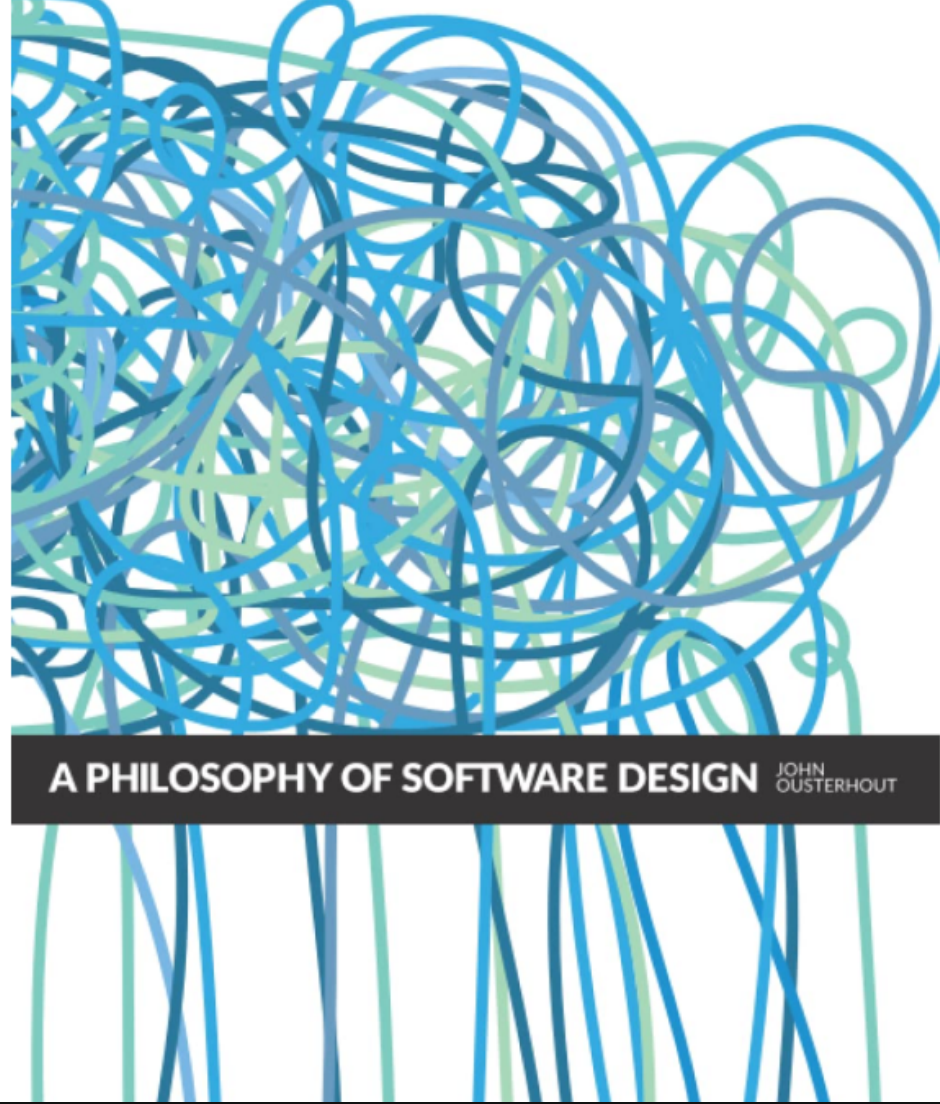
実践的
モデル

アーキテクチャが
同時に進化する

から学んだ
チームと
Spotify
Netflix
Google
Amazon

A Philosophy of Software Design

- システムが複雑になるメカニズムと、複雑性をもたらす問題、それを解決するための方法について解説している
- システムが複雑になるというのは、どういう状況を指しているのかの具体例が表記されていて、それを読むごとに、どこかで見たことあるわーみたいな共感があって面白い
- 複雑性を理解すれば逆にシンプルな設計ができる的な本



Clean Craftsmanship

規律、基準、倫理

- **規律、基準、倫理**の3つのレベルに分けて、プログラマー（and プログラマーのマネージャー）はどうあるべかのボブおじさんなりの哲学が書かれている
- デバッガーを今すぐ捨てるとか、完成しているコードにテストコードを書くからつまらないとか、企業がQAチームを設置しているのはプログラマーが仕事をしないからとか、過激な内容も記載されているが、実際は芯をくった内容だったりもして読んでいて面白い

Clean Craftsmanship

クリーンクラフトマンシップ Clean Craftsmanship

規律、基準、倫理

Robert C. Martin 著
角 征典 訳

より良く働き
生産性を高め
自分が書いたものに
誇りを持つ！

ASCII

世界一流エンジニアの思考法

- 米マイクロソフトで働く日本人エンジニアが著者
- 著者が一般的なエンジニアの視点で、米マイクロソフトシニアエンジニアと仕事して自身の思考回路の違いを実際にあったエピソードを交えて解説している
- 一般的には称賛されるような思考こそ、実は落とし穴がある的な本
- 読み物としても普通に面白い
- Audibleでも配信しているので手軽に聞ける



リファクタリング

既存のコードを安全に改善する

- 逆に一周してマーティン・ファウラーの名著を読ん
でみた
- リファクタリングの基礎/哲学について解説してい
る
- リファクタリングの原初の本なので、このリファク
タリングって、どんな効果があるの？みたいな疑問
を持ったことがある人にオススメ
- いろいろな技術書で名前が挙がることが多いので教
養と一読しておくとか解像度が上がるかも

リファクタリング

既存のコードを安全に改善する

第2版



まとめ

- OOUlを理解しているフロントエンドのコードのまとまり自体の認識が深まり変更に強いコードが書けるようになる
- スクラム、DevOps、チームトポロジーを読むと開発のサイクルを回し続けるために必要になる要素を理解できる
- A Philosophy of Software Designとリファクタリングは純粋にプログラミングとして勉強になる
- Clean Craftsmanship、世界一流エンジニアの思考法は哲学的な視点もありメンタルモデルの構築に役に立つ

ご清聴ありがとうございました 🎉