

『Tailwind CSS実践入門』 出版記念基調講演

いかにして「ユーティリティファーストへの入門書」ができたか



pixiv Inc.
Chiaki KUDO / @f_subal

pixiv



@f_subal

著者

自己紹介

- 2016年4月 新卒入社
- 2018年から pixivFACTORY の開発
- フロントエンドエンジニア → テックリード → エンジニアリングマネージャ → プロダクトマネージャ
- 2020年～2022年デザインシステムの開発を兼任
 - 2022年に charcoal として OSS 化
 - デザインシステム初期メンであり、OSS化プロジェクトの当事者
- 2024年に『Tailwind CSS実践入門』を出版

Tailwind CSS

実践入門

工藤智祥 [著]

Tailwind CSS

デザイナーと同じ言葉でマークアップをする
ユーティリティクラスという「契約」の力

- ユーティリティファーストでマークアップをする方法
- 設定ファイルの書き方とテーマのカスタマイズ
- ユーティリティクラスを用いたコンポーネントの設計
- 独自のプラグインをどう実装するか
- デザインシステムを構築するときの考え方

エンジニア
選書



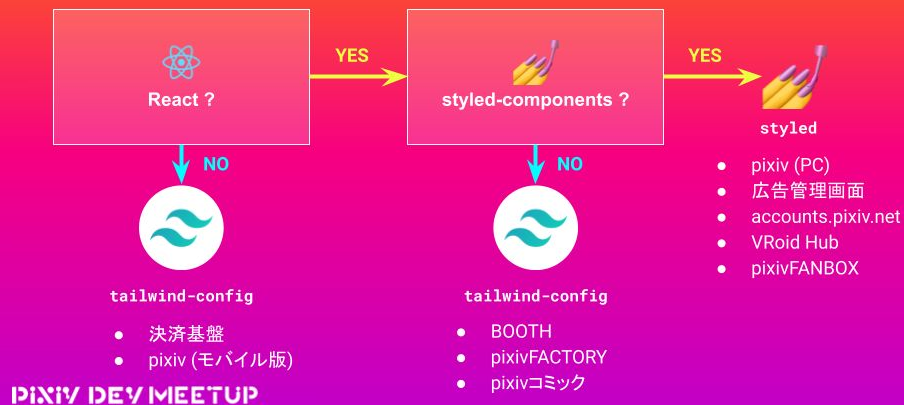
技術評論社

Tailwind CSSとの出会い

デザインシステムの 技術選定に悩んでた

- プロダクトが10個以上ある
- 技術スタックが全部Reactとは限らない
- その上で統一的なデザインシステムを作りたい
- **そんなところに Tailwind CSS がハマった**
- pixiv inside やカンファレンスでも何度かお話した内容

あなたにピッタリのユーティリティを選ぼう



<https://inside.pixiv.blog/2022/03/31/120000>

Tailwindに出会う前のCSSへのスタンス

- CSS in JSよりはCSS Modulesを好む
- CSSに近い、SassやPostCSSを好む
- BEMなどの設計論はあまりやらない
 - stylelint をガチガチにしてクラスの名前規則は特に設けない
- 「ガイドラインとしてのデザインシステム」に懐疑的
 - lint にない規約は信用しない
 - Sass変数の管理はコスパ悪い派



藤秋

@f_subal · フォローする



font-weight、定数を用意したところで人間はそんなもの使わないので、stylelint で特定の数字以外を指定したら怒ったほうが良いのではって気持ちがある

午前0:54 · 2019年12月22日



12



返信



ポストへのリンクをコピー

Xでもっと読む

powered by <https://tweet2image.vercel.app/>

CSSのレビューに対する課題感

＼ところで会場に質問です／

Q. ふだん開発チームで
コードレビューをしてる方？

…良いですね、では

Q. CSSのコードレビューを
他のファイルと同じぐらいの
正確さでやっている方？

CSSのレビューに対する課題意識

- 多くの開発者はCSSのコードレビューができない！
 - ※ このスライドは会場で思ったより多くの手が挙がった場合は破綻します
- CSSはwrite-onlyな言語になりがち
 - 書ける人は多いが、ちゃんと読める人が（読もうとする人も！）少ない
 - 読み書きの意志や能力にものすごく非対称性の高い言語
- 他のソースコードと同じぐらいの精度でCSSをレビューできるチームは多くない
 - しかし本来これはダメな状態
- 本来あるべきコードレビュー → 「このflexboxの書き方だとモバイルで崩れそう」
- 現実のコードレビュー → 「クラスの命名規則がルールに沿っていませんよ（CSSの中身はだれも見えない）」

「Tailwind? 私はCSS書けますし…」

- あなたがCSSを書けるかは問題ではない!!!!
 - 正しくは「チームがCSSをレビューできるか」
- 対比されるのは素のCSSではない
 - 比べるべきは stylelint とかだ
- 許可リストの Tailwind CSS vs 拒否リストの stylelint とかなら意味のある対比



藤秋

@f_subal · フォローする



私は CSS を書けますが、過去の CSS を消すために Tailwind を入れますという姿勢になってきた

午前0:27 · 2020年5月26日



15



返信



ポストへのリンクをコピー

Xでもっと読む


powered by <https://tweet2image.vercel.app/>

Q. Tailwind がレビューしやすいですって？

- A. クラスの多さに気を取られてるとそう思うかもしれない（人間の気持ち）
 - が、「他のファイルが上書きしているかも…」と考えなくて良いのは最高
 - それに「ダメな記述を見分けるルール」は結構シンプル。機械的にできる
- ルール1: Arbitrary Values -[...] があったら警戒する（4章と9章を参照）
- ルール2: モディファイアが少なかったら警戒する（やや主観的）
 - hover: や active: のないボタンはたぶん色の考慮が抜けてる
 - レイアウトに画面幅のモディファイアがなかったらレスポンスを忘れてる
- 機械にわかりやすいコードは人間もレビューしやすいを地で行くのがTailwind CSS
 - 早く人間の気持ちをやめよう！！

本に何を書くか

書くにいたる経緯

- 2020年後半にはTailwind CSSをすっかり日常的に使うようになってた
- 2023年2月WEB+DB PRESSで特集を  _yuheiy さんが書かれていた（私ではないです）
- その後書籍版の企画があがったが、いろいろあって_yuheiyさんから私を紹介いただいた
- 2023年3月頃から企画が始動



ターゲット読者層を決める

- 誰に向けて書くか？初心者？中上級者？
- 明らかに私に期待されてそうな役割 = デザインシステムでの活用法
 - もうこの時点で初心者向けではない
 - HTML/CSS自体を知らない人にこの話ができない
- フロントエンドエンジニアに絞るか？
 - これは No !!!!!
 - デザイナもマークアップエンジニアもサーバーサイドの人も読んでほしい
 - なぜならTailwind CSS自体がそのぐらい広いターゲット層のライブラリだから
 - React 前提の説明もあるが、それは仕方がない（最小限かつ疎にする）

「対戦相手」を決める

- 通底する二項対立があると本全体の議論が見やすくなる（仮想敵を置くのは大事）
 - 例: OOUI本の「タスク指向 vs オブジェクト指向」とか
 - 私の場合は「ユーティリティファースト vs セマンティックCSS」など
 - 必ずしもどちらかを悪者にする必要はないが、対立構造が見えるようにする
- 例1: 「Tailwind CSSは短くかけるのが良いところ」と言ってる人
 - 初心者だが、対決する相手ではない。彼らを成長させるのが狙い
- 例2: 素のCSSやPostCSSがあればデザインシステムはつくれると言ってる人
 - もしそうなら私はデザインシステムでこんなに悩んでない（反論したい）
- 例3: コンポーネント集がデザインシステムの必要十分条件だと思っている人
 - みんなあんだだけBootstrap上書きしたりMUIのsx使ったりしてるのに？

本を書くときのスタンス

差分ではなくストーリーを書く

- **技術ブログは「差分」を書くことが許される**
 - 最新情報を追ってる人は前回からのアップデートを知れば良い
 - フロントエンドの流れが速くて複雑に見えるのは、実際に色々起きてるのもあるが、みんなが「**前回との差分**」をしゃべっているから
- **本には「前回との差分」がない**
 - 界隈の全体像を、歴史的な流れの解釈を、一本のストーリーを提示する必要がある
- **Tailwind CSSは前提となる背景が多いので余計に意識した**
 - 結果的に第1章と第2章がすごい長さになったけど、まあそういうことです

「思想書」として書く

- フレームワークの入門書が扱うのは主に「歴史」と「思想」である
 - 作者の意思決定と世の中の変化、その間の論理の破綻のなさこそが重要
 - それを踏まえて「～な状況で〇〇する方が良い」と読者に提示するのがゴール
- テクノロジーを扱っているが、**歴史書か思想書だと思って書いたほうが上手くいく**
 - 本屋さんで「理工書」のコーナーに置かれるけど、**でもやるんだよの精神**
 - これは技術記事をZennやQiitaに書くときも同じこと
- 仕様や使い方の説明は「事実」なので正しいが、これだけでは新規性に乏しい
 - そうするのは仕様書や公式ドキュメントに任せる方が良い
 - 一番マズいのは「**内容は公式の事実しか言っていないのに文体だけエモい**」文章
 - **これの真逆に行くことが大事**

大変だった部分

本当は周辺ライブラリに詳しくなかった

- 私は「React・Tailwind CSS・clsx・React Ariaがあれば全部書けるじゃろ💪」
という発想だった
 - なので cva とか shadcn-ui とか prettier-plugin-tailwindcss とか有名どころをめちゃめちゃスルーしてました（すみません）
 - 6章を書く過程でキャッチアップをし、サンプルコードを書く前に何とか
- ポジティブに考えると「どのライブラリが必須でどのライブラリが好みの問題か」
の嗅覚はあった
 - なぜ私はこのツールをスルーしてたのか、ちゃんと言語化すれば大丈夫
- ここはレビュアーのみなさまにアドバイスをもらった面が強いです（多謝）

触ってない技術について堂々と語る方法

- ピエール・バイヤールは読書とは内容を知ることではなく本の「位置づけ」「本同士の関係」を理解することと説いた
- これは技術も同様に、ライブラリを知ることはライブラリ同士の関係を知ることと言える
- もちろん最終的には触りました
- ちなみに私はこの本は読んでます

Comment parler des livres que l'on n'a pas lus?

読んでいない本 について 堂々と語る方法

ピエール・バイヤール

Pierre Bayard

大浦康介 訳

Yasusuke Oura

ちくま学芸文庫

本当にCSS設計の歴史に疎かった

- 名前のついたコーディング規約を覚えるのにもともと苦手意識があった
 - 私のCSS設計に対するスタンスは最初に話した通り
- BEM以外よく分かってなかったし、その理解も結構間違っていた（ごめんなさい）
 - @_yuheiy さんその節はありがとうございました
- 代わりにCSSの技術的な発展史は語れることが多かったので、第1章がああなった
- 実は「**文章ベースのガイドラインを信じない人向けのCSS本**」という読者像がある
 - 第9章は暗にデザインシステム（≒しくみベース）と文章ベースの規約の二項対立がある

第4章の見積もりがマジで無理だった

- 執筆開始は2023年4月頃から
- 基本「1ヶ月に2章ずつ、書ける章から順に」を進めてた
- 元々4章でクラスを説明した後6章でコンポーネントの話を書くのがキレイかなーと思っていた
- が、4章の終わりが不確定すぎて一番最後に書くことになった



藤秋

@f_subal · フォローする



「Tailwind CSSの各クラスと修飾子について説明する章は他よりちょっと長くなりそうで～、5万から10万文字ぐらいになるかな～」→ 14万3千文字に...

午後5:49 · 2023年8月10日



16



返信



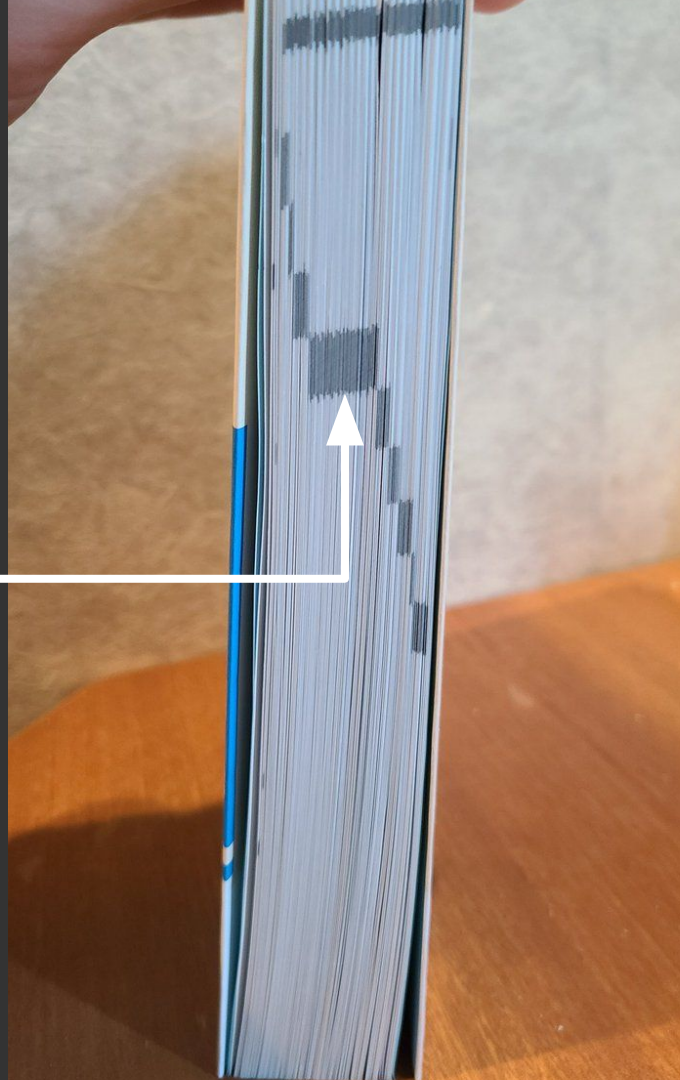
ポストへのリンクをコピー

Xでもっと読む

powered by <https://tweet2image.vercel.app/>

(ちなみに他の章は3万字ぐらいが中央値です)

第4章
Tailwind CSSで
マークアップする



文章の書き方に関する小ネタ (時間あったら)

かっこいいルビを入れる

- かっこいいルビには実用上のメリットがある
- この日本語とこの外来語が同じ意味であることを余計な説明なしで明示できる
- 例: 「プラグブル (pluggable)」という単語が初心者に易しくないと言われた
 - Tailwind CSSの用語じゃないし説明は省きたいな…文章のリズム悪くなるし
 - よっしゃルビにしよう、かっこいいし (大事)

プラグブル
実は着脱可能なプラグイン群によって実現されています。

モディファイア
後者はそれを表現する修飾子を示す (「p-4 クラスに画面

傍点を活用する

- 日本語にはイタリックがない
- しかし**強調の手段が太字しかない、というのは不便だ**
 - 多用するとブログっぽくなりすぎるので避けたい（いかがでしたか？感）
 - 注意喚起のためにイタリックで強調、みたいなことは日本語でもやりたい
 - 下線もなんか違うと思ってる（下線は読者が引くものだ）
- Tailwind本ではMarkdownでいう ****** を太字、 **_** を傍点に割り当てる方針を採った
 - **私は一般に `<em lang="ja">` は傍点で表すのが良いと思っている（思想）**

設定ファイルによるカスタマイズは、必ずしもやらねばならないというわけではありま

執筆をChatGPTに補助させる

- 厳密な定義のありそうな単語を不用意に使うのを避けたい
- 検索で自信が持てないとき一旦ChatGPTに聞く
- これだけで答えがわかることはマシなので追加の調査は必要だが（何で検索すればいいか分かる）
- その他、執筆に便利な GitHub Actions の生成とかにも使った



フィボナッチ数列の要件に、初項が1であることは含まれますか？
含まれる場合、初項が1ではないが、 $a[n] = a[n-1] + a[n-2]$ を満たす数列を総称するかわりの呼び方はありますか？



一般的に、フィボナッチ数列は最初の2つの項が1である数列を指します。つまり、初項が1であることはフィボナッチ数列の要件の一部です。

一方、初項が1でないが、再帰的な関係式 $a[n] = a[n-1] + a[n-2]$ を満たす数列は、フィボナッチ数列とは呼ばれません。このような数列を独自に識別するための総称的な用語はありませんが、"Fibonacci-like sequence"（フィボナッチに似た数列）や"generalized Fibonacci sequence"（一般化されたフィボナッチ数列）と呼ばれることがあります。

改めてありがとうございました



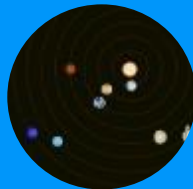
@laco2net



@okuto_oyama



@potato4d



@ushiro_noko



@petamoriken



@_yuheiy

レビュアーの皆さま、編集の久保田様、
告知に協力いただいたピクシブの皆さま
…And you !

Any Questions?