

CSS設計

の前準備

今回の概要

BEMやOOCSSなど世に広まっているCSS設計手法にも多少は触れますが、どちらかと言うと具体的な設計モデルを決めるまでの過程のお話になります。複数名で新規サイト立ち上げやリニューアルのコーディングをする際にお役に立てたら幸いです。

アジェンダ

1. 良いCSSって？
2. CSSってなんで破綻しやすいんだろう
3. 困った過去の事例たち
4. 具体的な設計に入る前に
5. 各CSS設計の概要
6. まとめ

良いCSSって？

「予測しやすい」「再利用しやすい」
「保守しやすい」「拡張しやすい」

グローバルスコープ

詳細度が
管理されてない

CSSってなんで破綻しやすいんだろう

マークアップに
依存している

親要素のスタイルが
子要素に継承

誰でも書けるから

困った過去の事例たち

cssの指定の仕方

- タグ(エレメント)指定
- 隣接セクタ・子セクタ指定
- 意味の無いid指定の多様
- スタイリングのためのclassに javascriptに関連する指定
- cssの詳細度が異常に高い

命名規則

- 命名規則の略語の使用可否
- ローマ字表記(ヘボン式/訓令式)
- キャメルケース・スネークケース・チェインケースの法則性

その他

- 影響範囲の調査が大変
- プロパティの順序に法則性なし
- 色指定(16進数・RGB・HSL)に法則性なし
- 波括弧“{}”の改行位置
- インデントがスペースとタブ混在
- ファイルによって改行コードが異なる
- 対象ブラウザが不明確
- 横並びにする手段が色々
- etc...

というわけで

具体的な設計に入る前に

話し合う 必要がある

- 設計するための基礎情報を知る
- 依頼元(所属先)の人員体制を知る
- 各メンバーと共通認識を持つ

設計するための基礎情報を知る

クライアントへのヒアリング内容例

- 「新規サイトの立ち上げ」or「既存サイトのリニューアル」どちらか？
- 対象の規模
- 全体のスケジュール
- 「静的ページ」or「動的ページ」どちらか？あるいは両方か？
- 作成するスタイルガイド(CSS設計)の活用範囲(外部委託の際の活用など)
- 既に構想を抱いている設計の有無

※CSS設計を依頼されるケースは大抵「新規サイトの立ち上げ」あるいは「既存サイトのリニューアル」のどちらかです。

また、「動的ページ」はエンジニアが関与するため、スケジュールに大きく影響します。

依頼元(所属先)の人員体制を知る

管理者またはクライアントへのヒアリング内容例

- コーダー以外がCSSを記述する可能性の有無(WEB担当者・ディレクター・デザイナー・サーバーサイドエンジニア)
- メインになる作業担当者(プロパー社員・派遣社員などのパートナー・外部委託)

※CSSに「関わる人」や「作業担当者から推測出来る稼働計画」は選定する設計方法や難易度に影響すると考えてます。

各メンバーと共通認識を持つ

PJメンバーとのブレスト内容例

- 現状のルール
- 現状の問題点と解決策
- CSS設計の理解度とそれに基づいたコーディング経験
- SassなどのCSSプリプロセッサの使用経験
- webpackまたはgulpなどのタスクランナーの使用経験

※CSS設計を提供するあたって「運用フェーズ」を考慮することは必須です。チームの技術レベルによってはOJTなどによって技術の底上げからスタートさせる必要があります。

各CSS設計の概要

	普及率	設計難易度	クラスの付け方	メリット	デメリット
BEM	高	低	シングルクラス	採用しているところが多い。 どこで使われているか分かりやすい。	class名が長くなる。 厳格なルール設計なので理解を間違えると破綻する。
OOCSS	高	高	マルチクラス	パーツの使いまわしがしやすい。 コード量を減らせる。	コードが複雑になる。 影響範囲を調べるのが大変。
SMACSS	高	中	マルチクラス	パーツの使いまわしがしやすい。 コード量を減らせる。 プレフィックスから用途が想像しやすい。	Sassを利用を前提としているので、 生のCSSだと恩恵を受けにくい。
FLOCSS	中	中	マルチクラス	公式ドキュメントが日本語。 ファイル・ディレクトリ構成が定義されている。 プレフィックスから用途が想像しやすい。	Sassを利用を前提としているので、 生のCSSだと恩恵を受けにくい。 ファイルが細分化されているため記述場所に悩む。
ECSS	低	中	シングルクラス	影響範囲がわかりやすい。	「再利用」を想定していないので必然的にコード量は増える。 事例が少ない。

色々あるけど一長一短です。

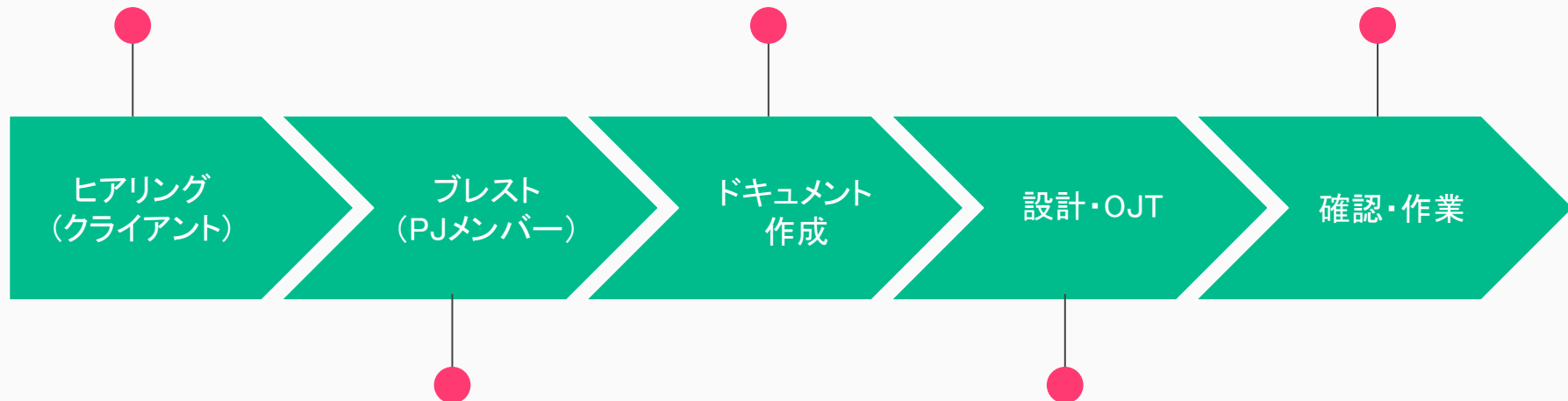
現場の状況・ニーズを踏まえた上で、
設計を選択し、必要に応じて拡張してください。

まとめ

- 目的
- 基礎情報
- 要望

- 作業計画
- 制作環境関連
- 技術関連

- 最終確認
- コーディング開始
- 外部への依頼



- 新旧のルール
- 問題点・解決策
- 技術レベル

- CSS設計(スタイルガイド作成)
- サンプル作成
- OJT(PJメンバー)

ご清聴ありがとうございました。